Machine Learning for Subgraph Extraction: Methods, Applications and Challenges

Kai Siong Yow   Ningyi Liao   Siqiang Luo   Reynold Cheng

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

香 港 大 學
THE UNIVERSITY OF HONG KONG

49th International Conference on Very Large Data Bases
Vancouver, Canada - August 28 to September 1, 2023

About the Presenters

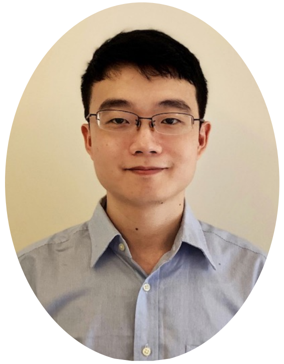
Reynold Cheng

Professor, HKU

ckcheng@cs.hku.hk

https://www.reynold.hku.hk

Data science, Big graph analytics, Uncertain data management



Siqiang Luo

Assistant Professor, NTU

siqiang.luo@ntu.edu.sg

https://siqiangluo.com

Graph algorithms/systems, KV systems, ML data management

Ningyi Liao

Ph.D Candidate, NTU

liao0090@e.ntu.edu.sg

https://nyliao.github.io

ML graph algorithms, Graph Neural Network

Kai Siong Yow

SASEA Fellow, NTU

kaisiong.yow@ntu.edu.sg

Graph theory, Data management, Computational mathematics

Outline

Introduction Siqiang Luo, 10 min

Community Search
Siqiang Luo 12 min

Community Detection
Ningyi Liao 10 min

Q&A 5 min

Maximum Common Subgraph
Ningyi Liao 12 min

Subgraph Isomorphism Counting
Reynold Cheng 12 min

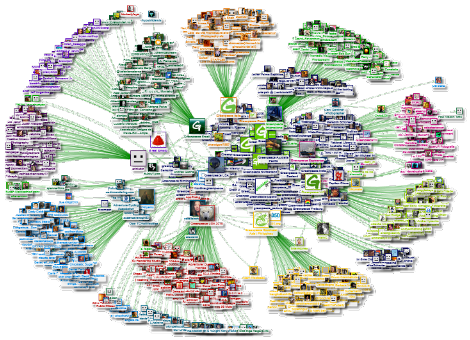
Conclusion & Future Directions Reynold Cheng, 10 min

Q&A 20 min

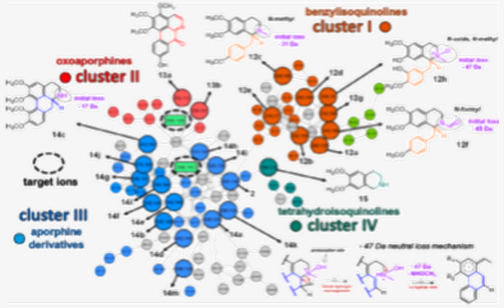
Outline

- Introduction Siqiang Luo, 10 min
- COMMUNITY SEARCH
- COMMUNITY DETECTION
- Q&A
- MAXIMUM COMMON SUBGRAPH
- SUBGRAPH ISOMORPHISM COUNTING
- Conclusion & Future Directions
- Q&A

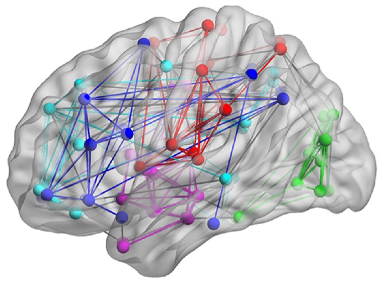
Graphs Are Everywhere



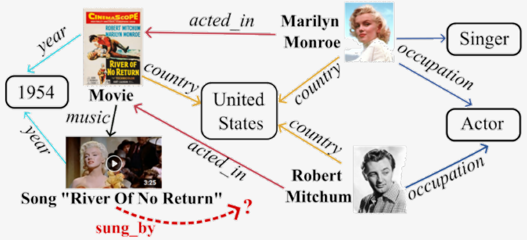
Social Networks



Molecular Networks

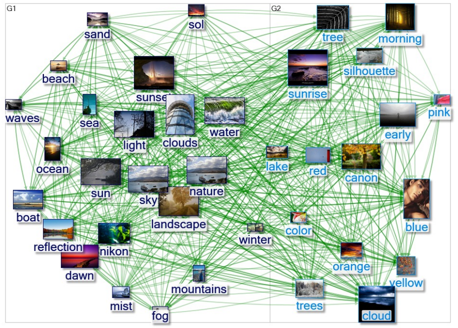


Human Brain Networks

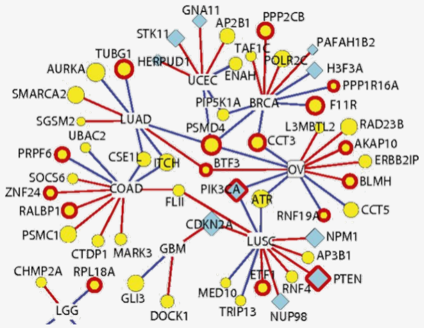


Knowledge Graphs

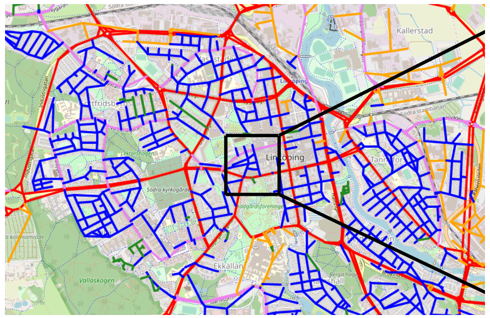
Transportation Networks



Tag Networks



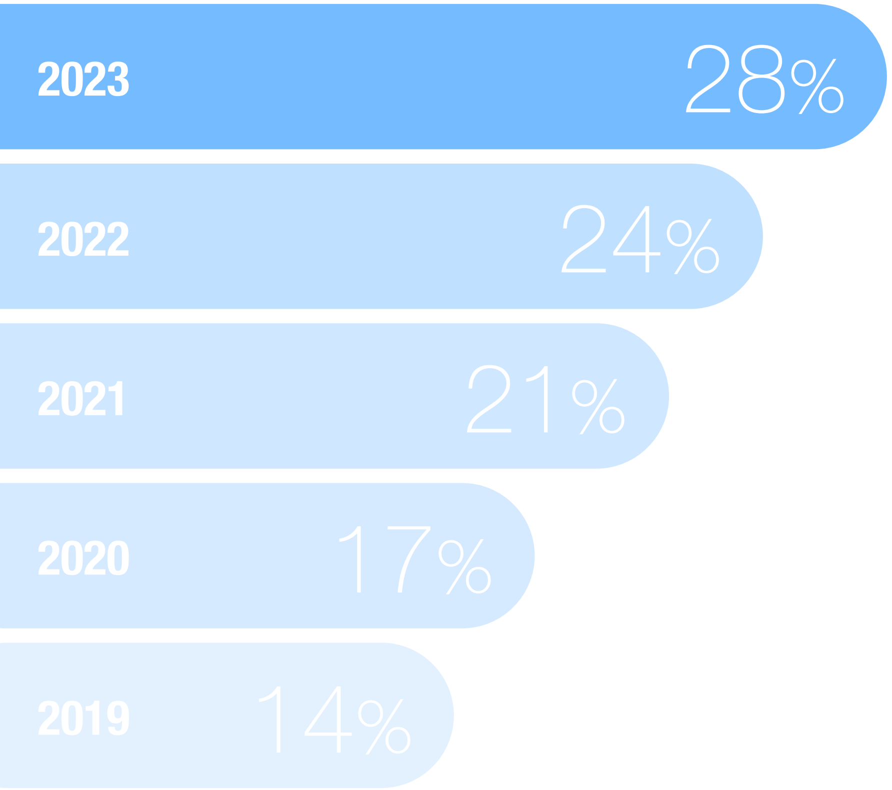
Biological Networks



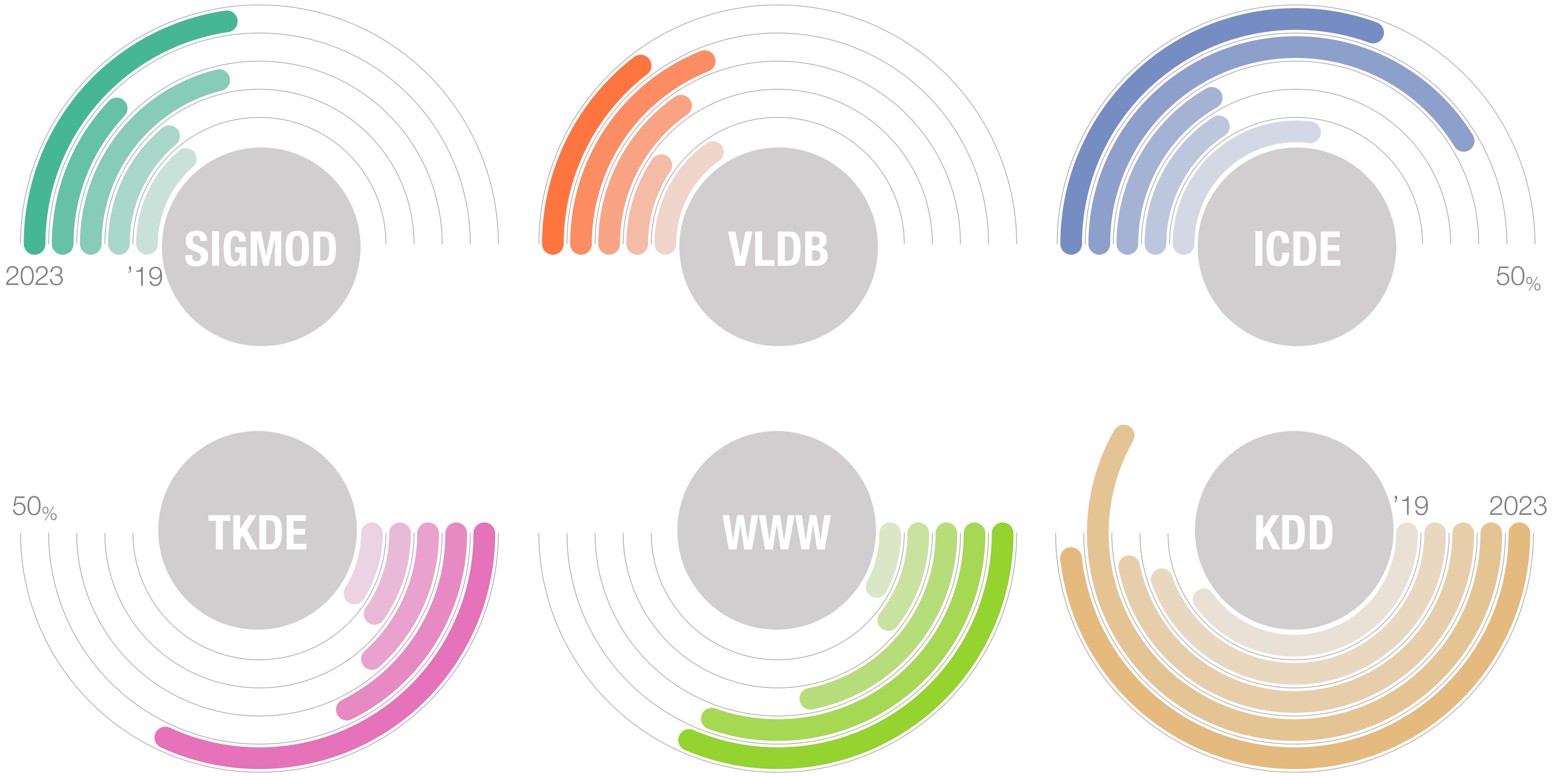
Road Networks

Images downloaded from the Internet.

Graph Research in Recent Years

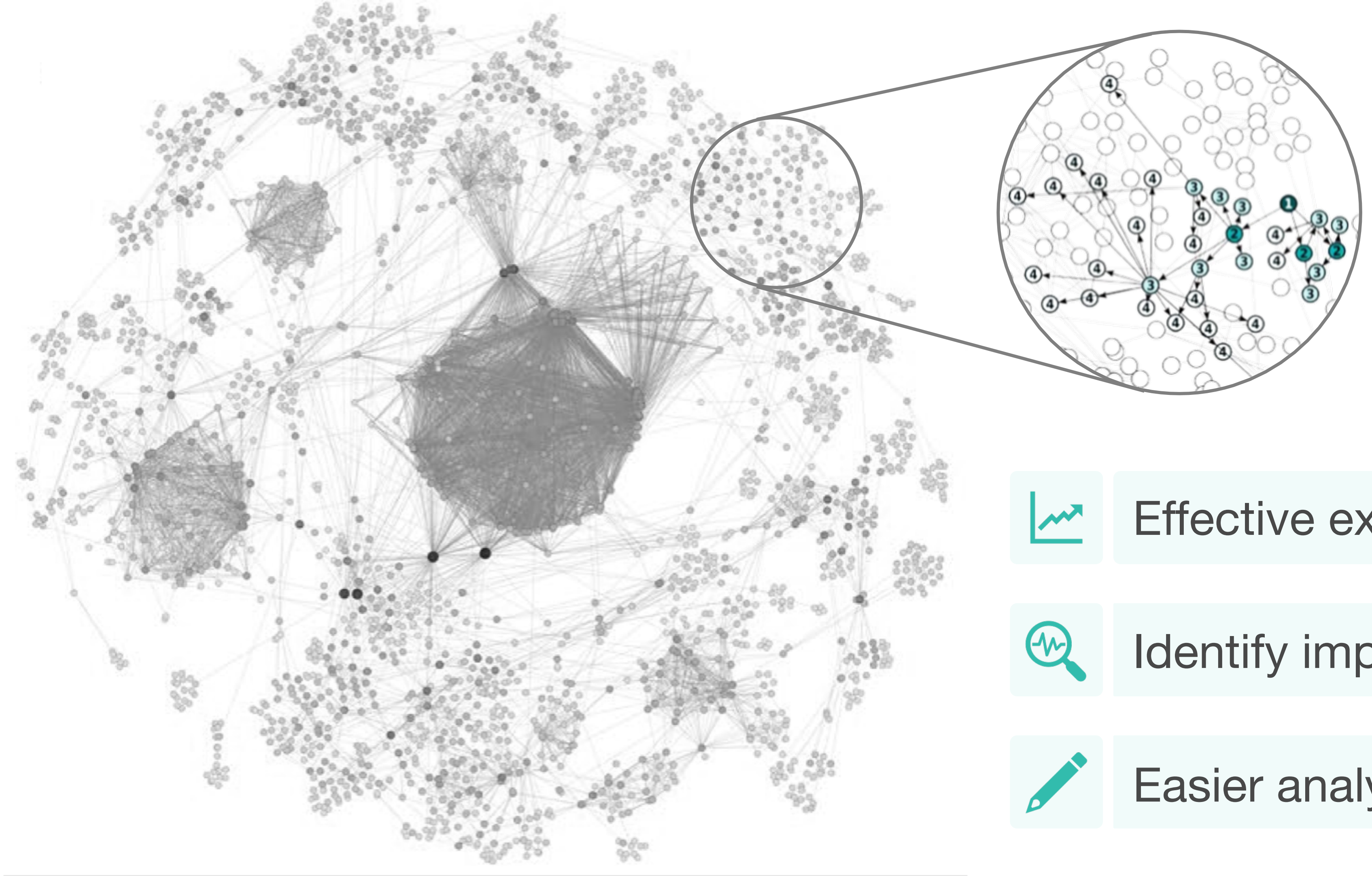


Graph research percentage
(in total)



Graph research percentage
(selected venues)

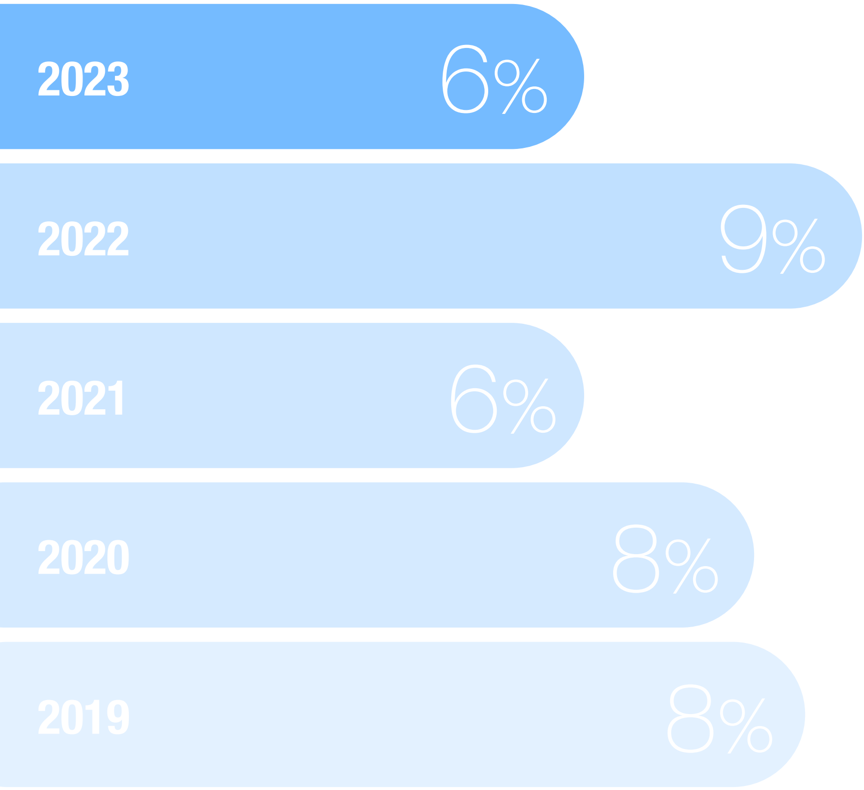
Subgraph Extraction – Why



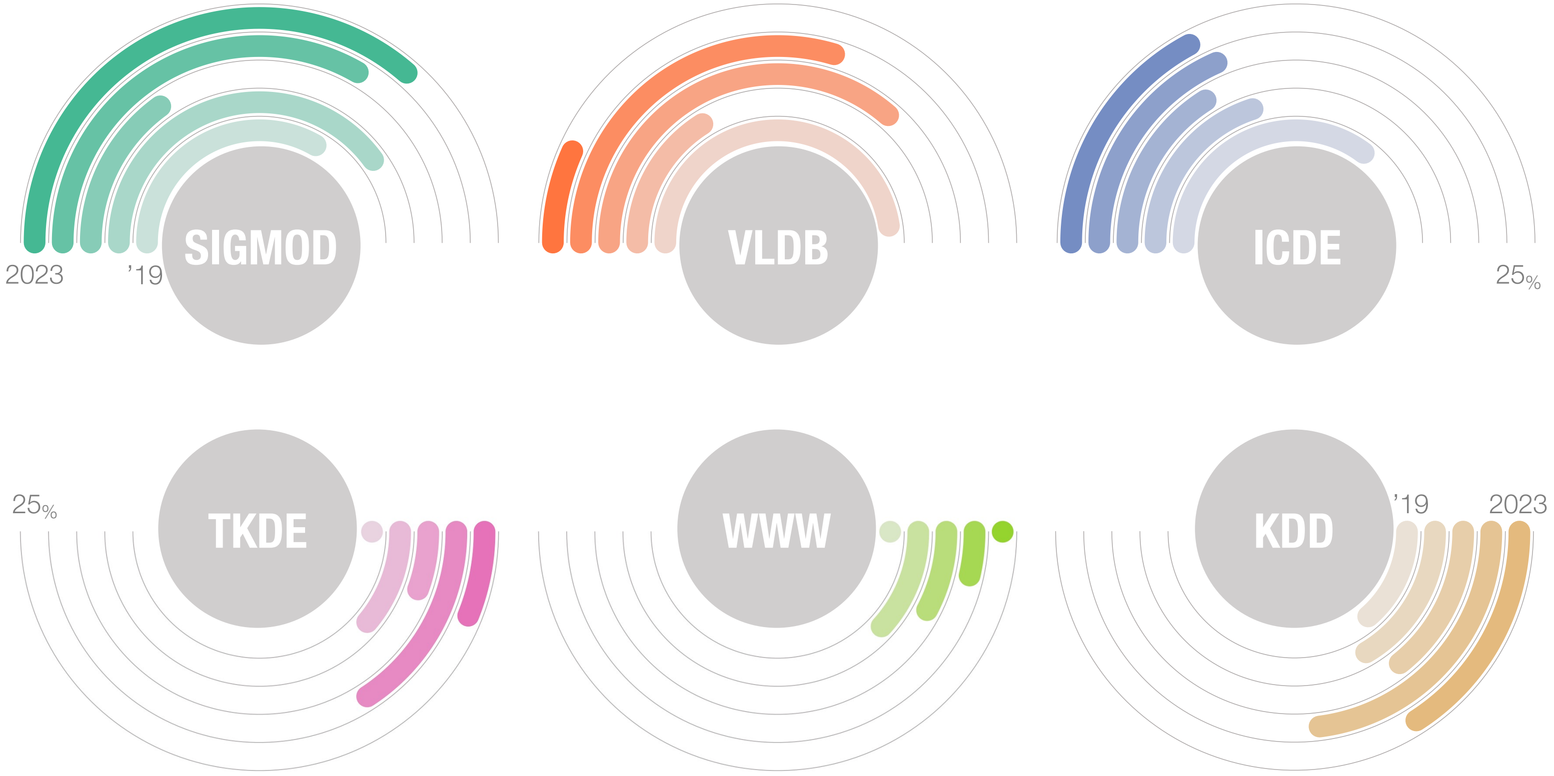
A sub-graph is a part of the original graph

- Effective exploration on large graphs
- Identify important structures
- Easier analysis and visualisation

Subgraph Research in Recent Years

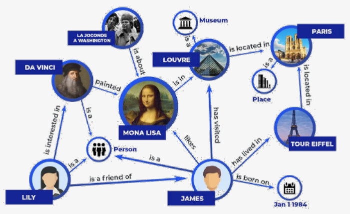


Subgraph in graph research
(in total)



Subgraph in graph research
(selected venues)

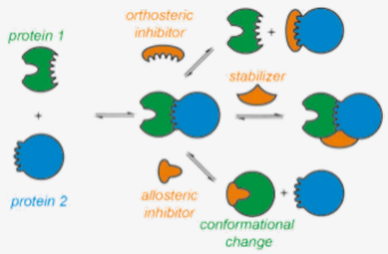
Subgraph Extraction is Widely Adopted



Knowledge Discovery
Knowledge Graph

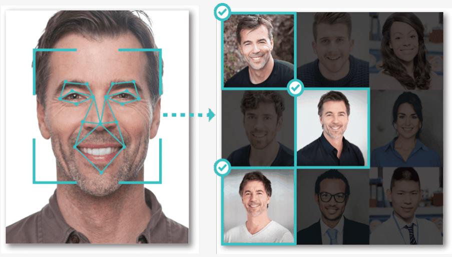
Advertisement Recommendation
E-commerce

Friend Suggestion
Social Network

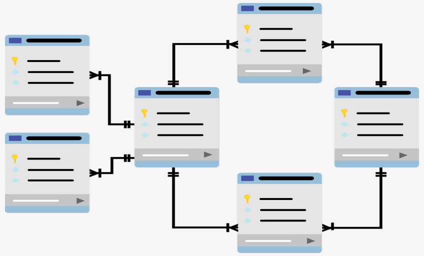


Drug Discovery & Function Analysis
Protein Interaction

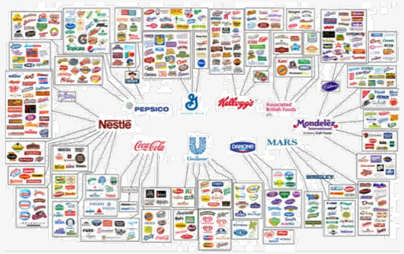
Subgraph Extraction



Facial Recognition
Face Landmark Image



Query Debugging
DBMS Schema



Pattern Discovery
Inter-firm network

Fraud Detection
Transaction Network

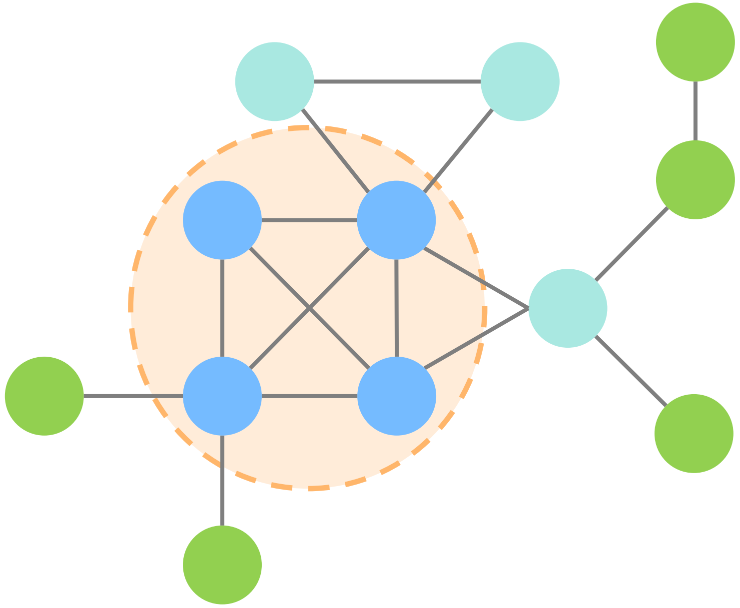
ML for Subgraph Extraction – Why

Limited Flexibility

- Predefined schemes are rigid when applied to varying scenarios

COMMUNITY SEARCH

COMMUNITY DETECTION



k-core

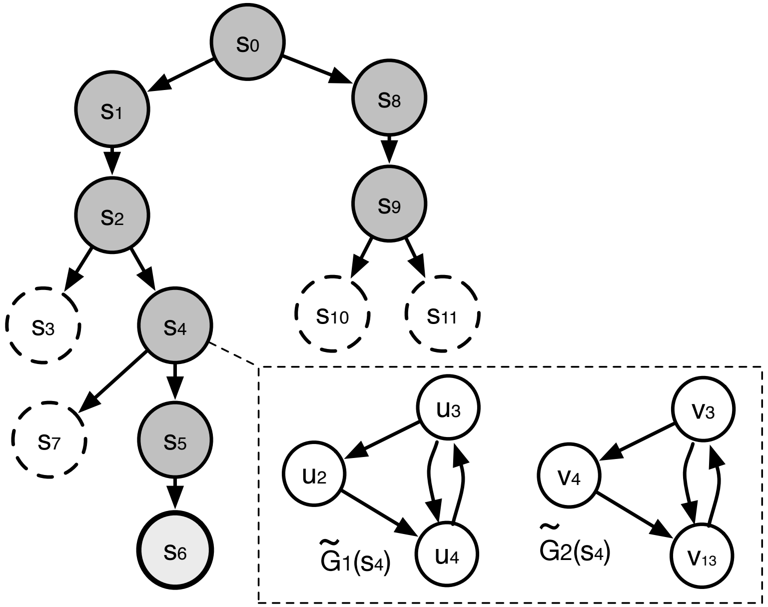
ML for Subgraph Extraction – Why

Limited Efficiency

- NP-hardness results in expensive overhead of algorithmic solutions

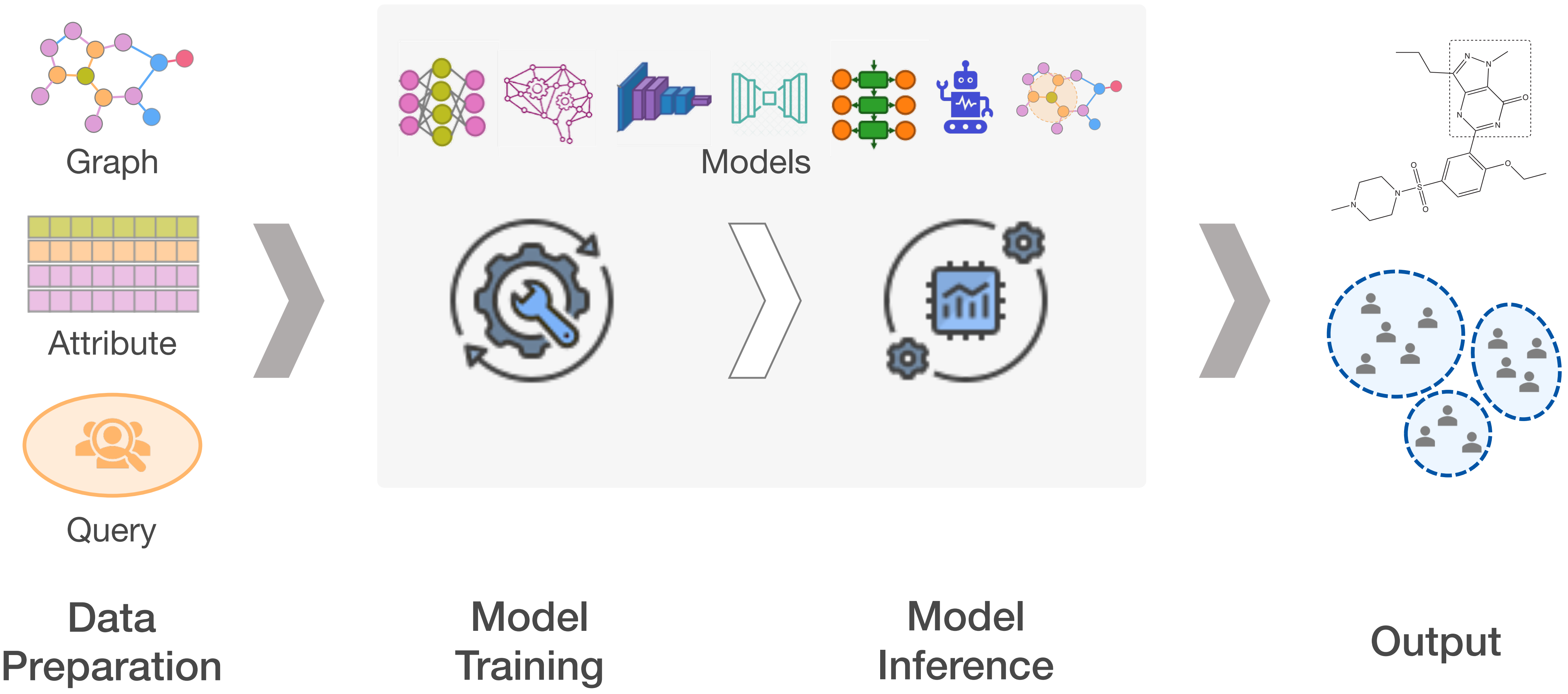
MAXIMUM COMMON SUBGRAPH

SUBGRAPH ISOMORPHISM COUNTING



A Search Tree

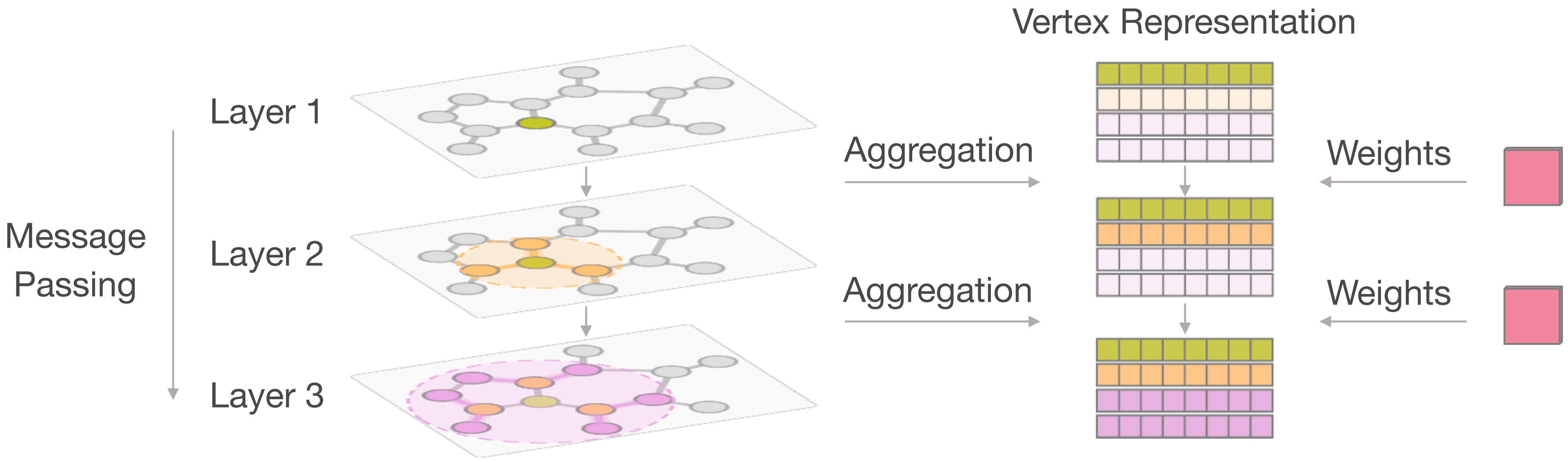
Common Graph ML Pipeline



Common Graph ML Approaches

GNN: Graph Neural Network

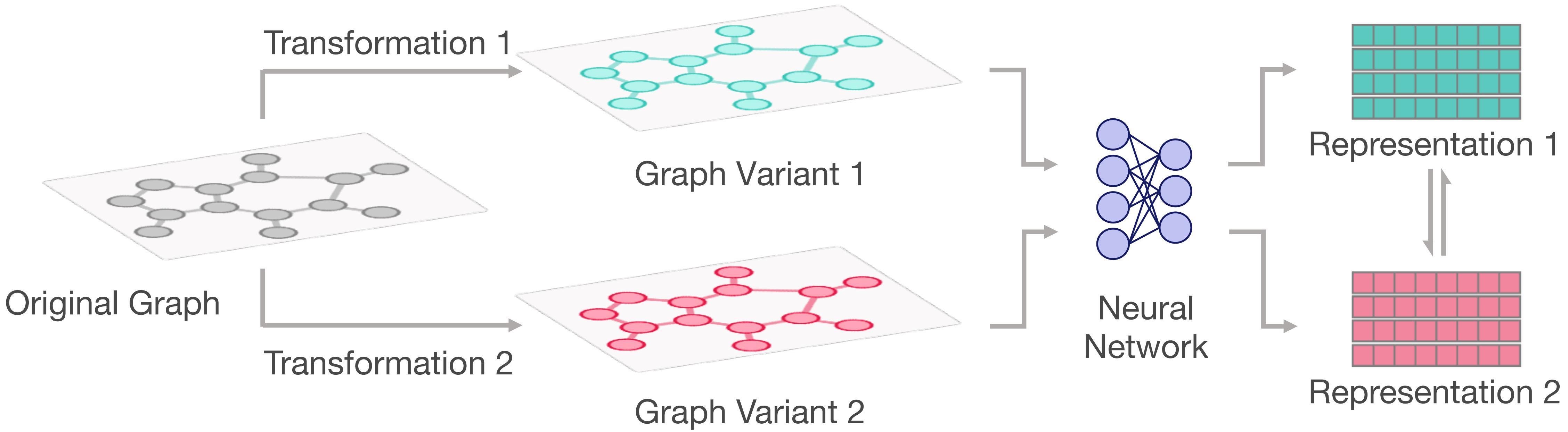
- Iteratively aggregates vertex neighbour information by learnable weights to learn representation



Common Graph ML Approaches

GCL: Graph Contrastive Learning

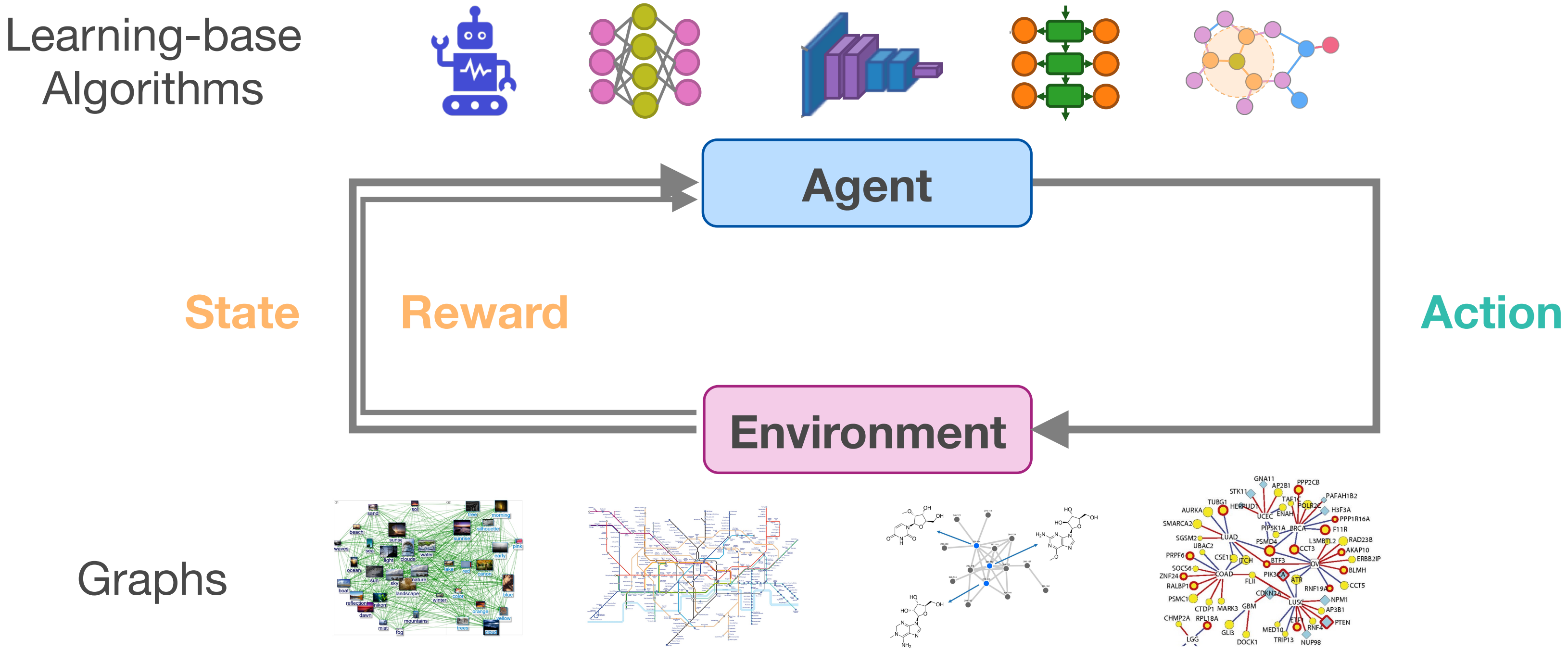
- A type of semi-supervised learning that generates and learns from similar and dissimilar graph variants



Common Graph ML Approaches

RL: Reinforcement Learning

- An agent interacts with the graph environment to learn to maximise the reward over a course of actions



Outline

Introduction

COMMUNITY SEARCH
Siqiang Luo 12 min

COMMUNITY DETECTION

Q&A

MAXIMUM COMMON SUBGRAPH

SUBGRAPH ISOMORPHISM COUNTING

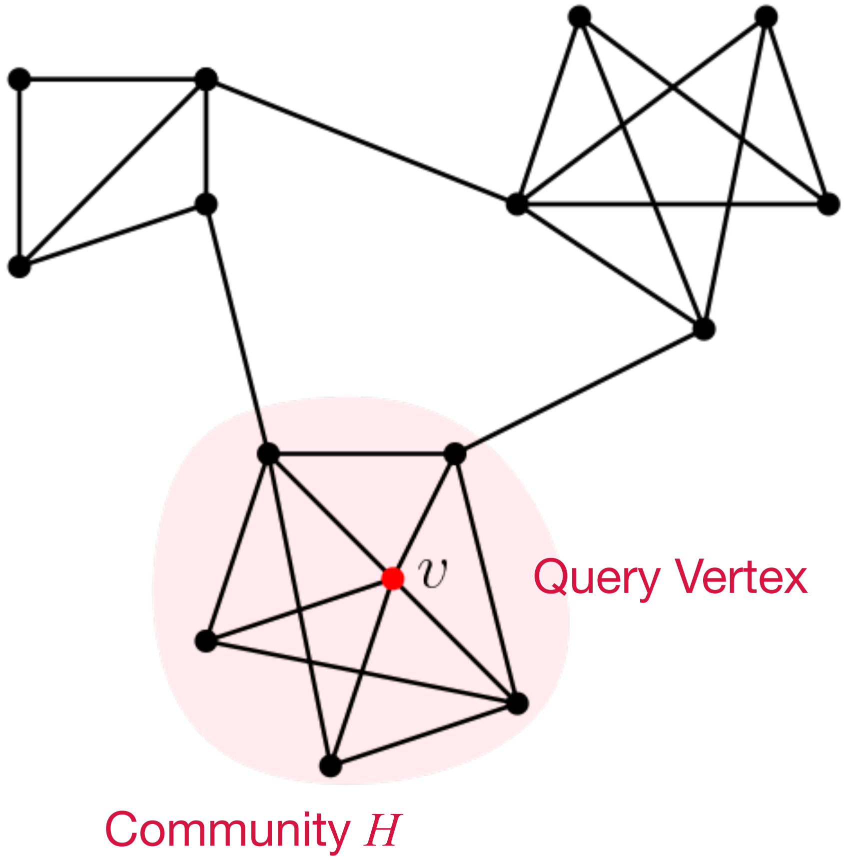
Conclusion & Future Directions

Q&A

Community Search

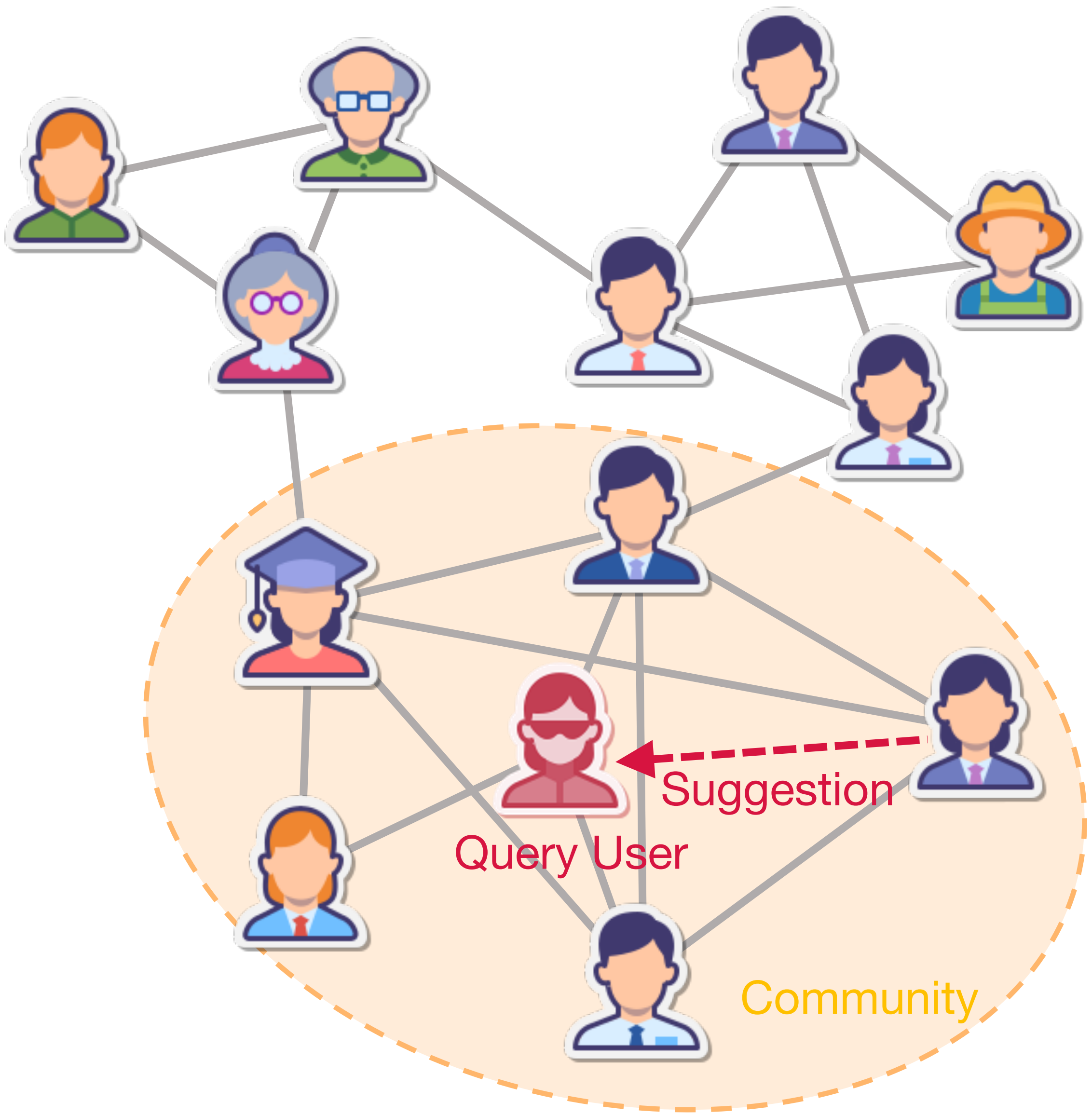
CS: Community Search

- Variant of Community Detection
- Deduce a subgraph H that
  - **Contains** a given query vertex v (or a set of query vertices)
  - **Satisfies** the cohesiveness and connectivity constraints

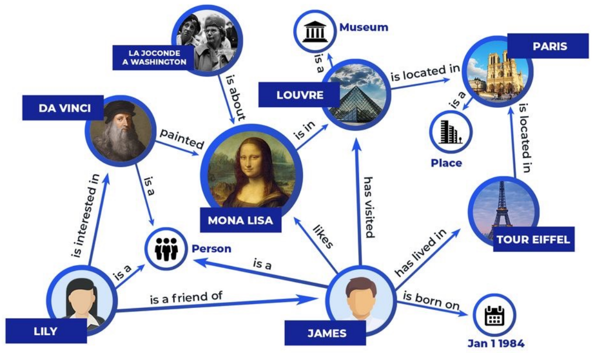


CS: The Applications

- **Graph:** social network
- **Vertex:** user
- **Edge:** friend connection
- **Query:** given user
- **Task:** Users tend to make friends within a same community. *How to search for a community that contains a particular user?*



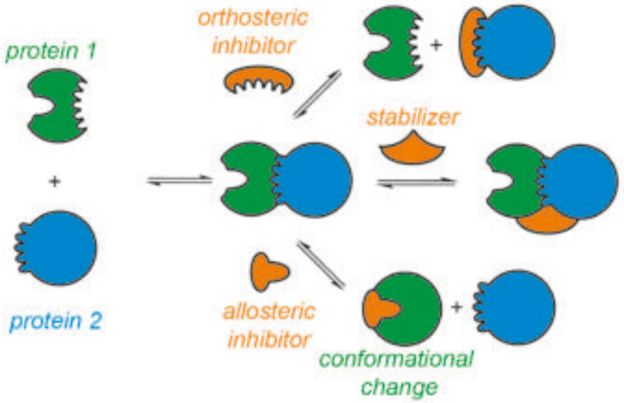
CS: The Applications



Knowledge Base

Graph: knowledge graph

Task: discovery new connections in an area



Protein Interaction

Vertex: protein | Edge: interaction

Task: discover functional ties between proteins

E-commerce

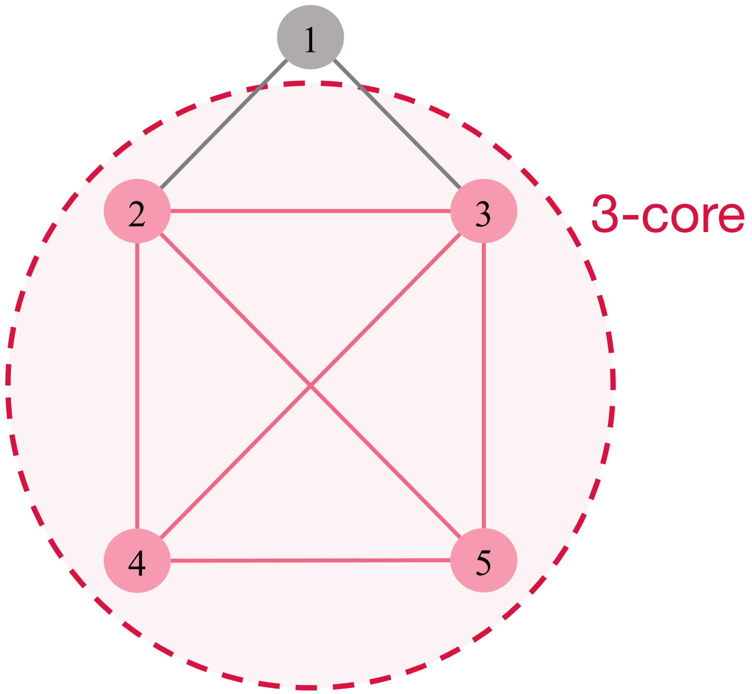
Graph: user community

Task: ad recommendation from other community members

Classical Metrics

***k*-core**
[KDD'10; SIGMOD'14; VLDB'16; VLDB'21]

A maximal connected subgraph H such that $\deg(v) \geq k$ for each $v \in H$



Classical Metrics

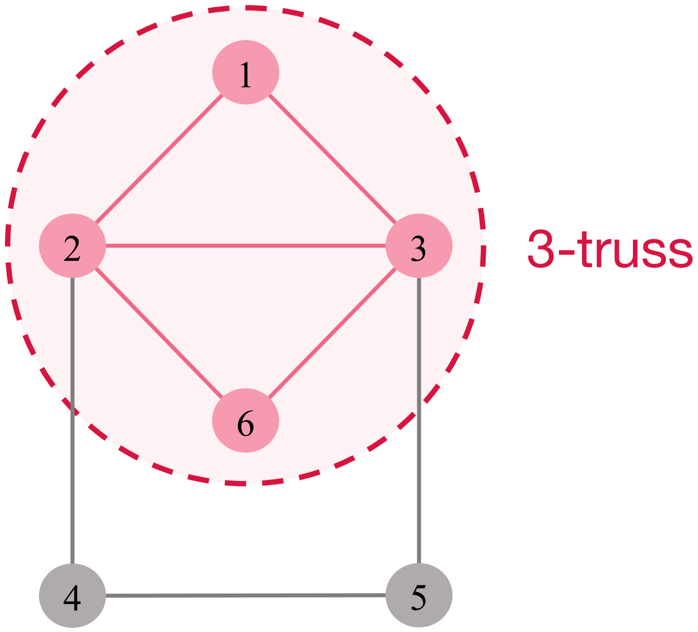
***k*-core**

[KDD’10; SIGMOD’14; VLDB’16; VLDB’21]

***k*-truss**

[SIGMOD’14; VLDB’15; VLDB’17; ICDE’21]

A maximal connected subgraph H such that every edge $e \in E(H)$ belongs to at least $k-2$ triangles in H



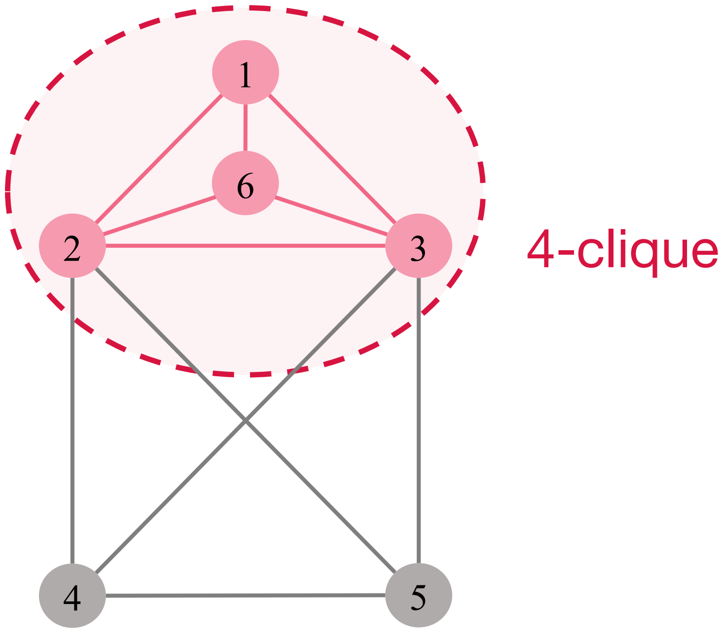
Classical Metrics

***k*-core**
[KDD'10; SIGMOD'14; VLDB'16; VLDB'21]

***k*-truss**
[SIGMOD'14; VLDB'15; VLDB'17; ICDE'21]

***k*-clique**
[SIGMOD'13; TKDE'17]

A connected subgraph H such that H is a complete graph of order k



Classical Metrics

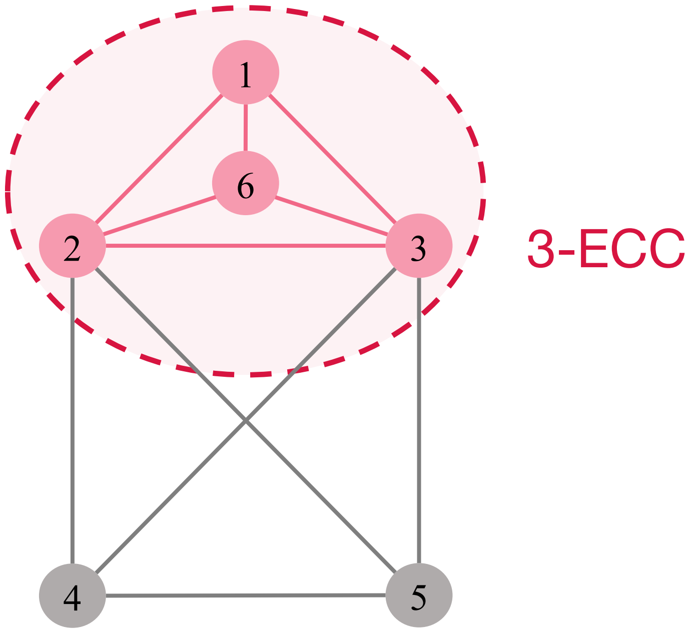
***k*-core**
[KDD'10; SIGMOD'14; VLDB'16; VLDB'21]

***k*-truss**
[SIGMOD'14; VLDB'15; VLDB'17; ICDE'21]

***k*-clique**
[SIGMOD'13; TKDE'17]

***k*-edge-connected component**
[SIGMOD'15; CIKM'16]

A connected subgraph H such that H remains connected if less than k edges are removed



Classical Metrics

***k*-core**
[KDD'10; SIGMOD'14; VLDB'16; VLDB'21]

***k*-truss**
[SIGMOD'14; VLDB'15; VLDB'17; ICDE'21]

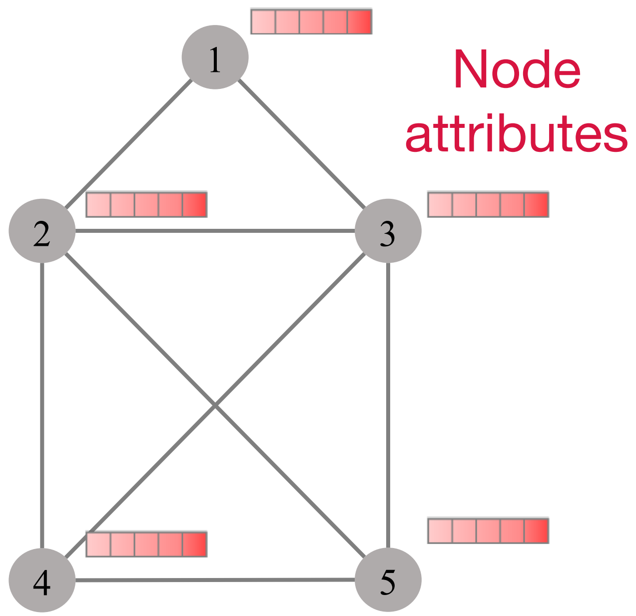
***k*-clique**
[SIGMOD'13; TKDE'17]

***k*-edge-connected component**
[SIGMOD'15; CIKM'16]

ACQ
[VLDB'16]

ATC
[VLDB'17]

Deterministic methods that apply on attributed graphs



Classical Metrics

k-core
[KDD'10; SIGMOD'14; VLDB'16; VLDB'21]

k-truss
[SIGMOD'14; VLDB'15; VLDB'17; ICDE'21]

k-clique
[SIGMOD'13; TKDE'17]

k-edge-connected component
[SIGMOD'15; CIKM'16]

ACQ
[VLDB'16]

ATC
[VLDB'17]

Lack of flexibility

Predefined patterns are rigid when applied to varying scenarios

Recent Learning Framework

Classical Method

Learning Method

Accurate Definition

Metrics such as k-core

Soft Information

p is in q's community

“Searching” the community

“Learning” the community model

Recent Learning Framework

ICS-GNN [VLDB'21]

Gao J, Chen J, Li Z, Zhang J. ICS-GNN: Lightweight interactive community search via graph neural network. PVLDB 2021.

- Yield high-quality communities with interactive labeling
- No predefined pattern needed

QD/AQD-GNN [VLDB'22]

Jiang Y, Rong Y, Cheng H, at al. Query driven-graph neural networks for community search: From non-attributed, attributed, to interactive attributed. PVLDB 2022.

- Model the attribute relations
- Process structure and attribute simultaneously

COCLEP [ICDE'23]

Li L, Luo S, Zhao Y, Shan C, Qin L, Wang Z. COCLEP: Contrastive Learning-based Semi-Supervised Community Search. ICDE 2023.

- Utilise graph contrastive learning
- Reduce the amount of training labels

ICS-GNN

Input query vertex

Gao J, Chen J, Li Z, Zhang J. ICS-GNN: Lightweight interactive community search via graph neural network. PVLDB 2021.

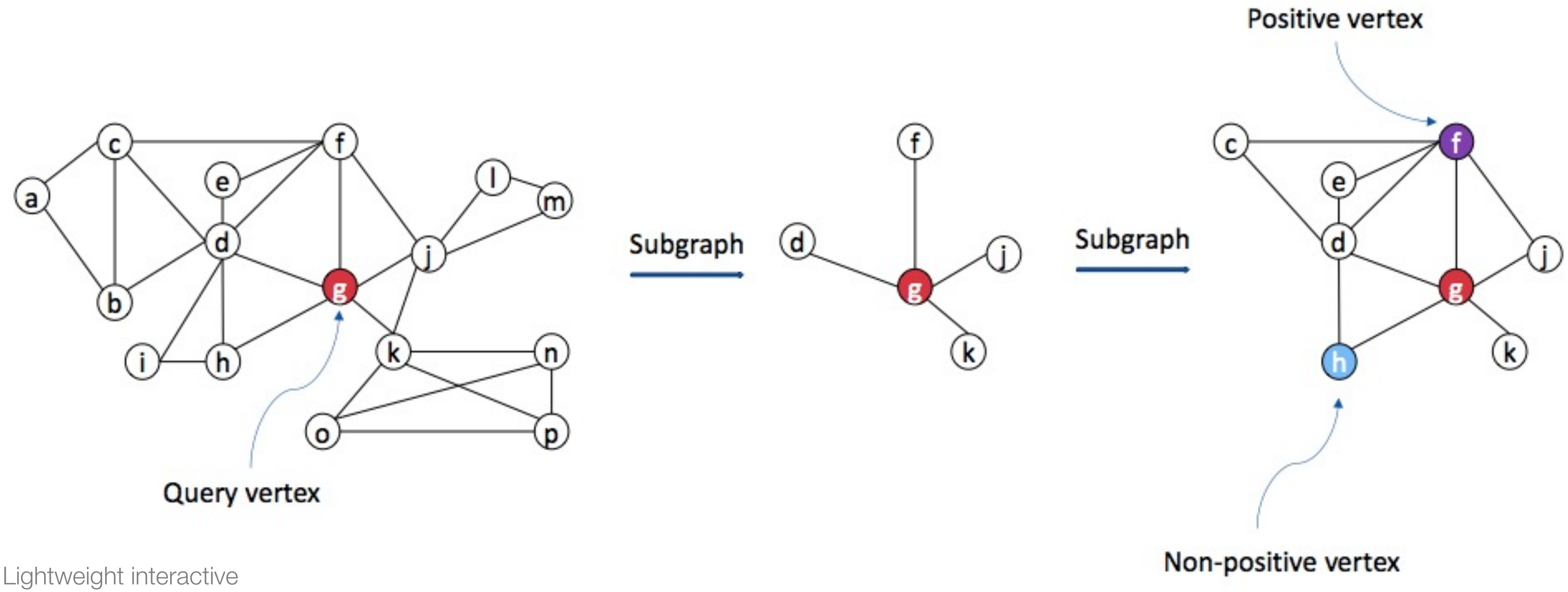
ICS-GNN

Iterative Learning & Search

Input query vertex

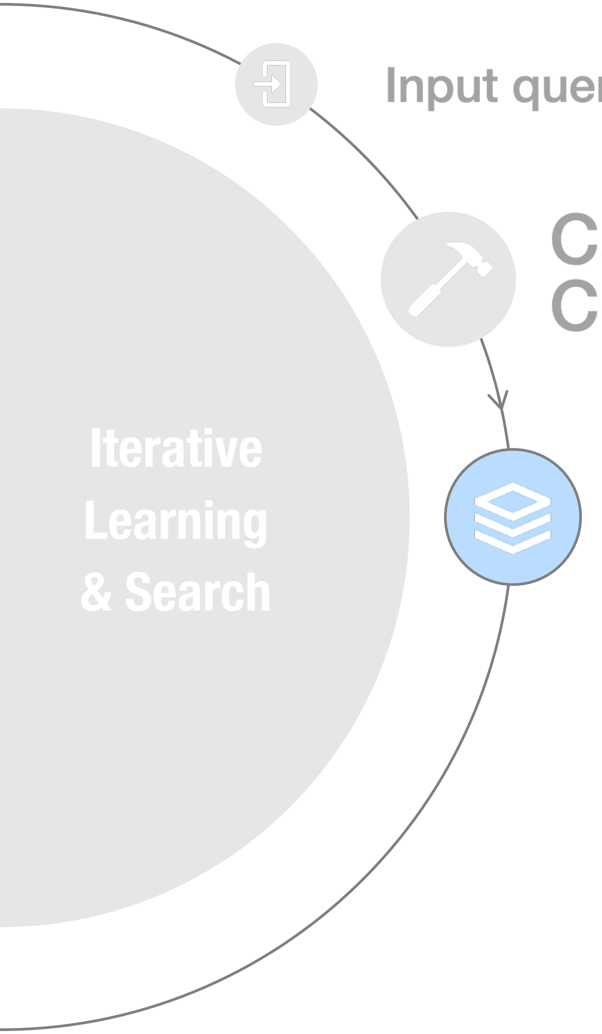
Candidate Subgraph Construction

- Partial edge enhancement strategy
- Locate useful vertices



Gao J, Chen J, Li Z, Zhang J. ICS-GNN: Lightweight interactive community search via graph neural network. PVLDB 2021.

ICS-GNN



Input query vertex

Candidate Subgraph Construction

- Partial edge enhancement strategy
- Locate useful vertices

GNN Training and Inference

- Output model that predicts the probabilities of nodes belonging to the community

Gao J, Chen J, Li Z, Zhang J. ICS-GNN: Lightweight interactive community search via graph neural network. PVLDB 2021.

ICS-GNN

Iterative Learning & Search

Input query vertex

Candidate Subgraph Construction

- Partial edge enhancement strategy
- Locate useful vertices

GNN Training and Inference

- Output model that predicts the probabilities of nodes belonging to the community

Community Discovery

- Find community with maximum GNN prediction score

Gao J, Chen J, Li Z, Zhang J. ICS-GNN: Lightweight interactive community search via graph neural network. PVLDB 2021.

ICS-GNN

Iterative Learning & Search

Input query vertex

Candidate Subgraph Construction

- Partial edge enhancement strategy
- Locate useful vertices

GNN Training and Inference

- Output model that predicts the probabilities of nodes belonging to the community

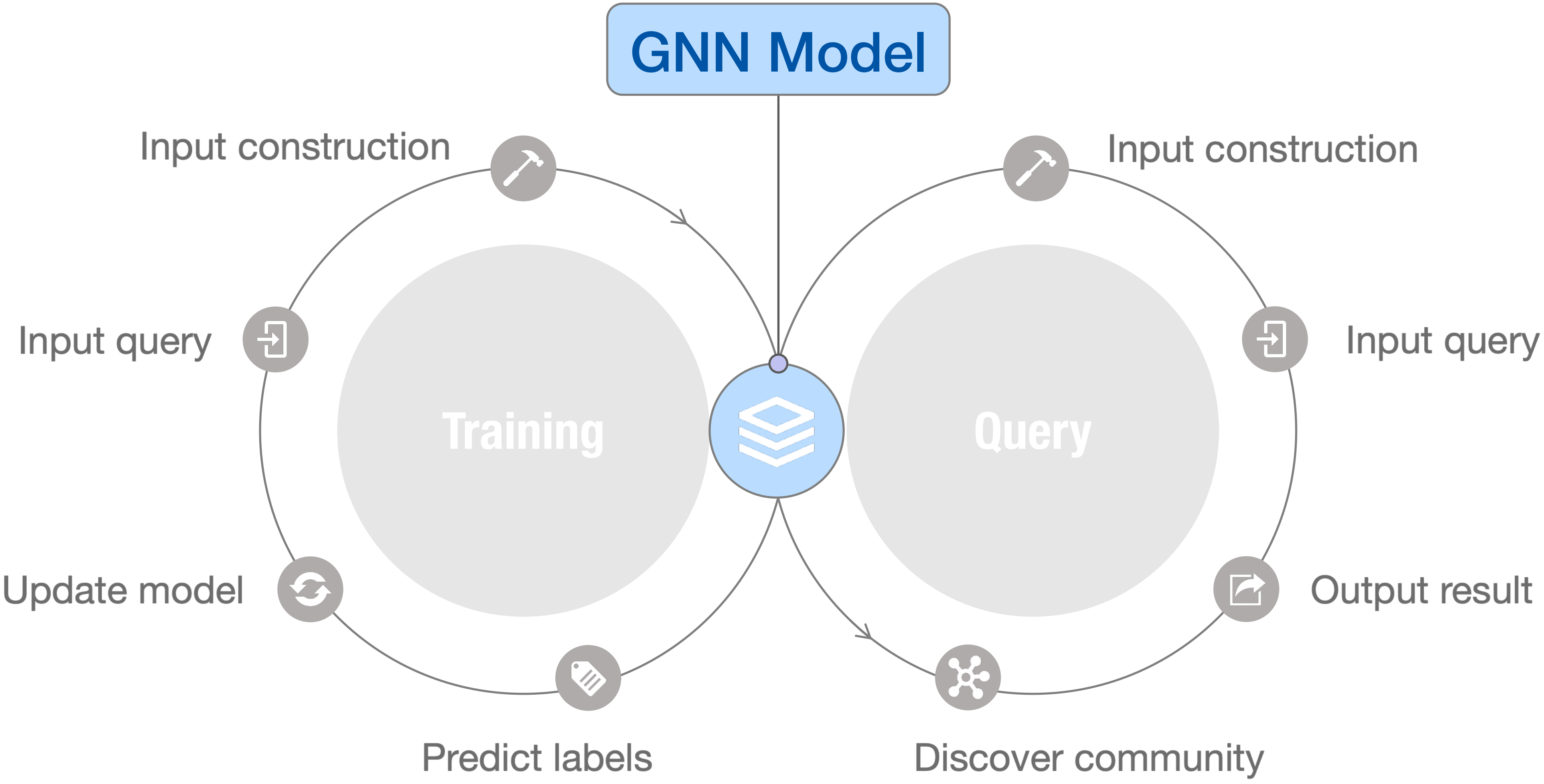
Community Discovery

- Find community with maximum GNN prediction score

Update labelled vertices

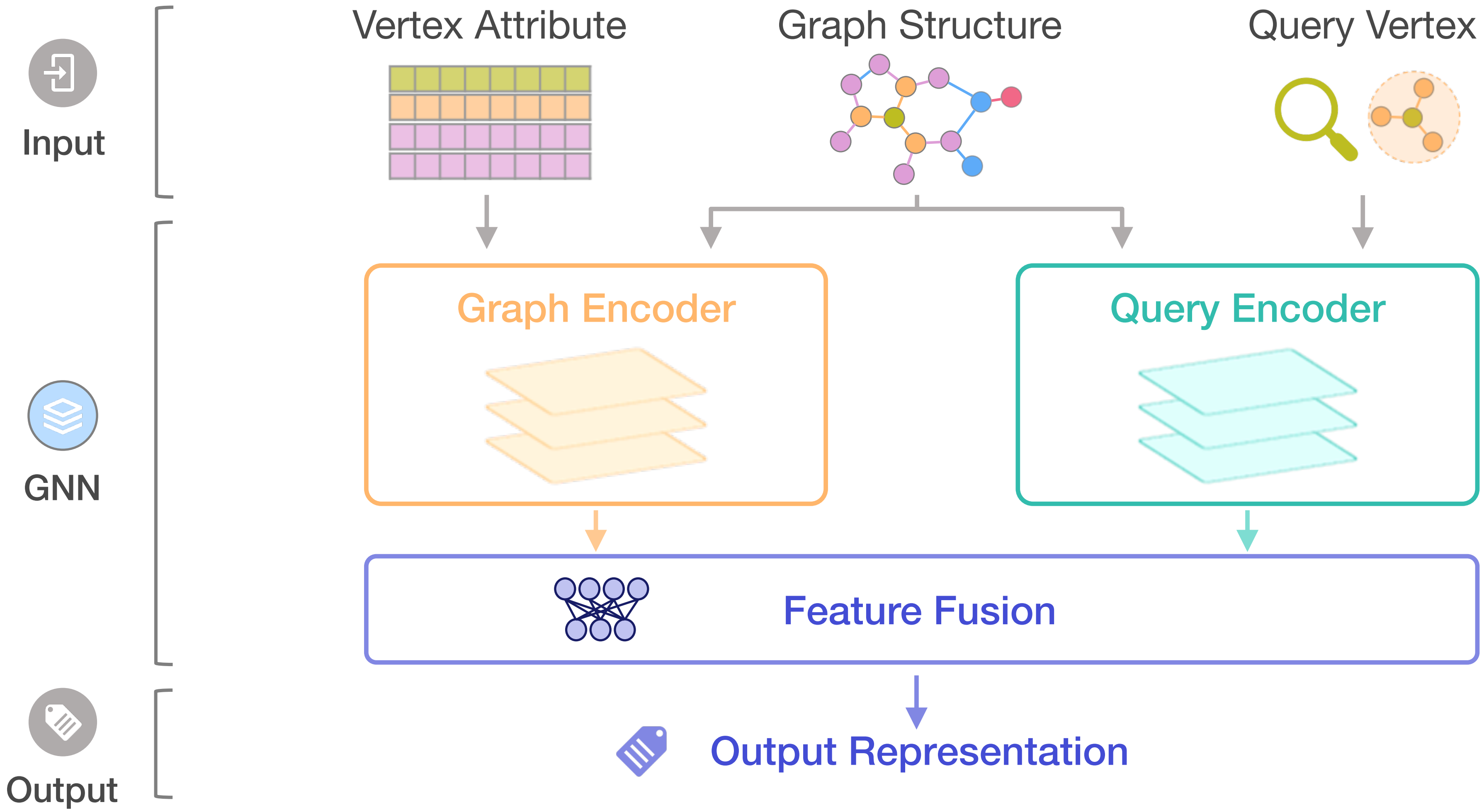
Gao J, Chen J, Li Z, Zhang J. ICS-GNN: Lightweight interactive community search via graph neural network. PVLDB 2021.

QD/AQD-GNN



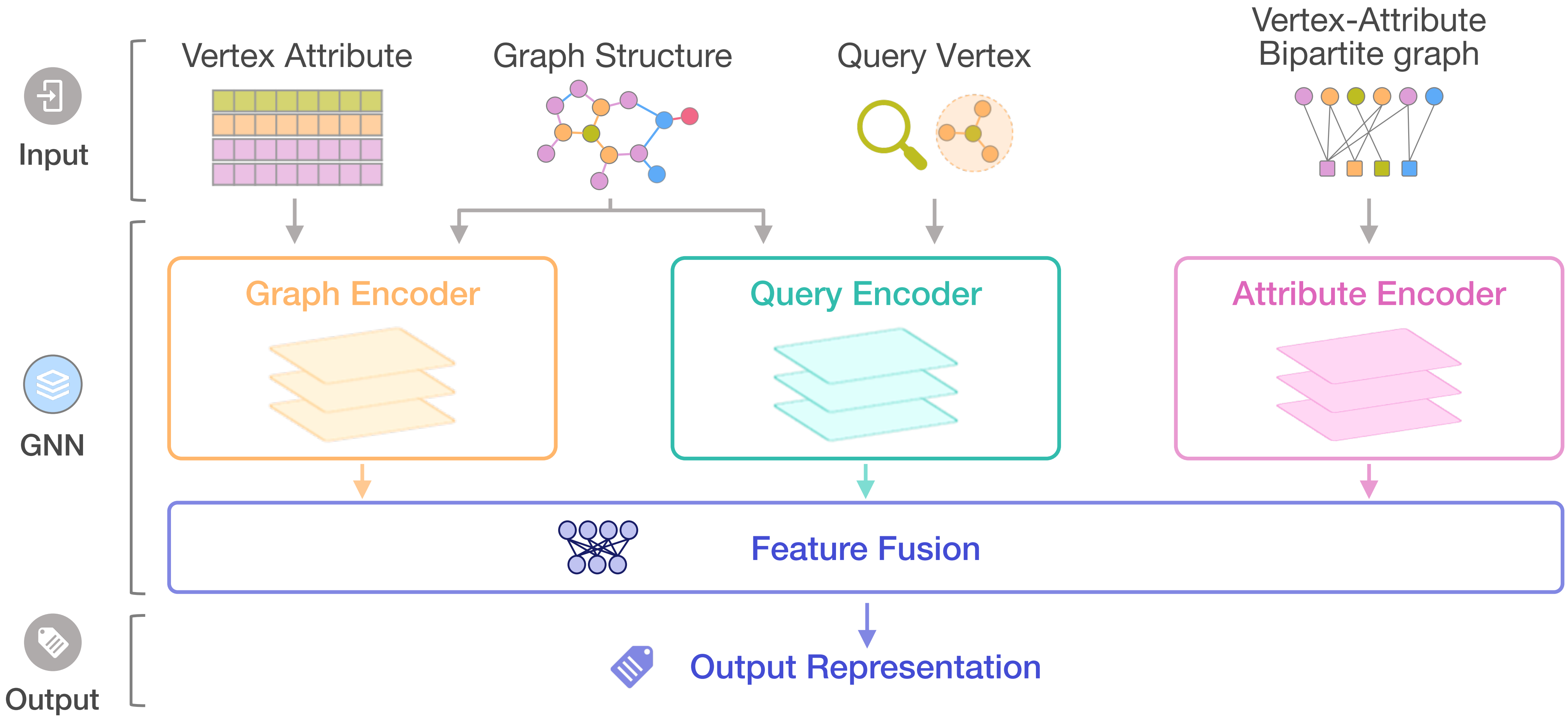
Jiang Y, Rong Y, Cheng H, at al. Query driven-graph neural networks for community search: From non-attributed, attributed, to interactive attributed. PVLDB 2022.

QD/AQD-GNN



Jiang Y, Rong Y, Cheng H, at al. Query driven-graph neural networks for community search: From non-attributed, attributed, to interactive attributed. PVLDB 2022.

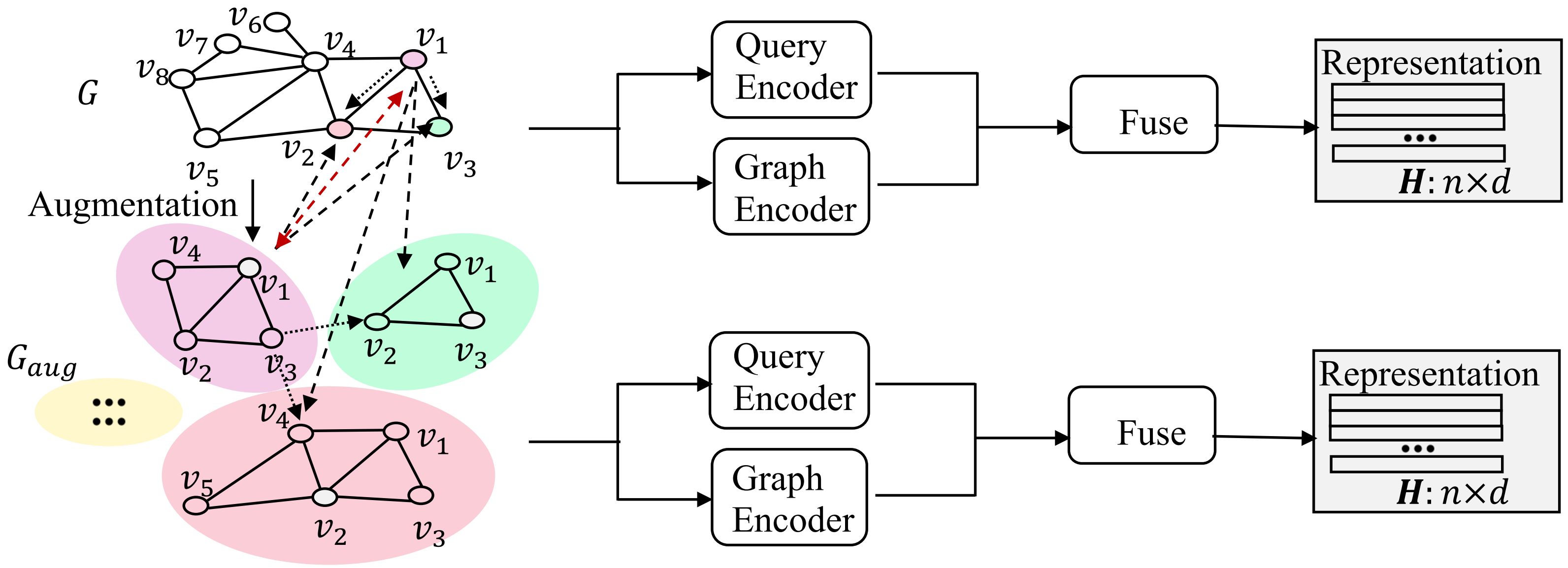
QD/AQD-GNN



Jiang Y, Rong Y, Cheng H, at al. Query driven-graph neural networks for community search: From non-attributed, attributed, to interactive attributed. PVLDB 2022.

COCLEP

How to reduce the demand of training labels?



Li L, Luo S, Zhao Y, Shan C, Qin L, Wang Z. COCLEP: Contrastive Learning-based Semi-Supervised Community Search. ICDE 2023.

CS: Summary

ICS-GNN

- Interactively explore and refine the community
- Trains GNN model for each query

QD/AQD-GNN

- QD-GNN: two-branch model that encodes information from both queries and graphs
- AQD-GNN: Extend by fusing attributes into the model

COCLEP

- Focus on reducing the label demands by using GCL

Outline

Introduction

COMMUNITY SEARCH

COMMUNITY DETECTION

Ningyi Liao 10 min

Q&A

MAXIMUM COMMON SUBGRAPH

SUBGRAPH ISOMORPHISM COUNTING

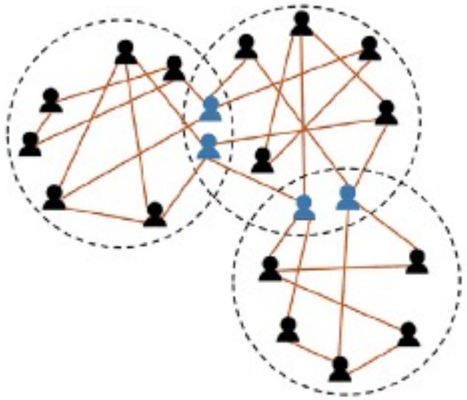
Conclusion & Future Directions

Q&A

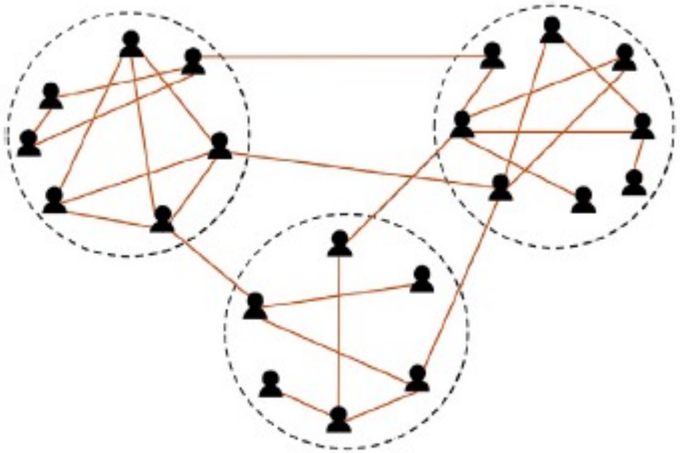
Community Detection

CD: Community Detection

- Partition a graph into a set of communities
- A community is a subgraph that satisfy cohesiveness and connectivity constraints
- Communities can be either disjoint or overlapping



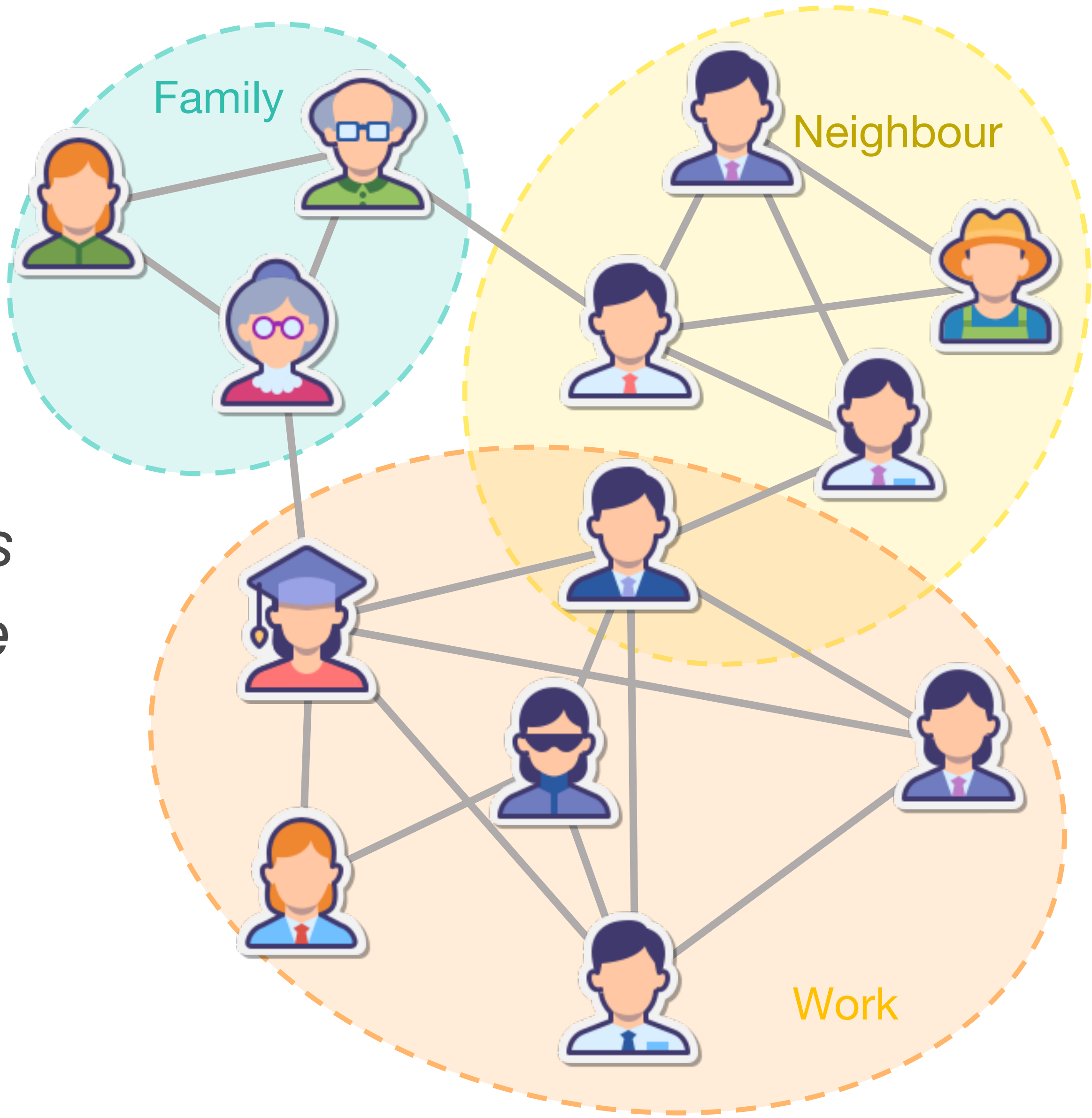
Overlapping Communities



Disjoint Communities

CD: The Applications

- **Graph:** social network
- **Vertex:** user
- **Edge:** friend connection
- **Task:** *How to detect communities containing similar users and close connections?*

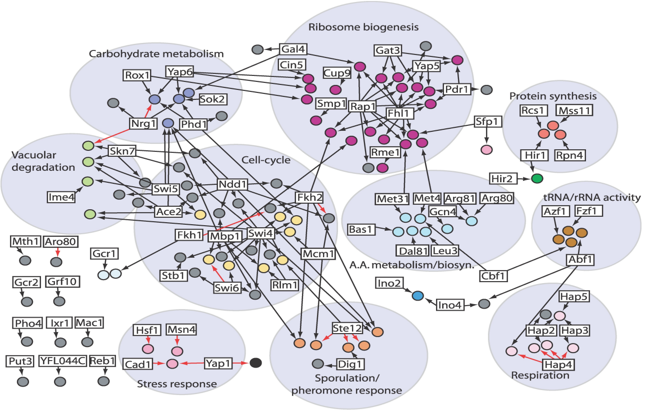


CD: The Applications

Friend Suggestion

Graph: social network

Task: suggest friendship in the same community



Biological systems

Graph: protein Interaction

Task: identify functional groups without prior knowledge

Fraud Detection

Graph: transaction network

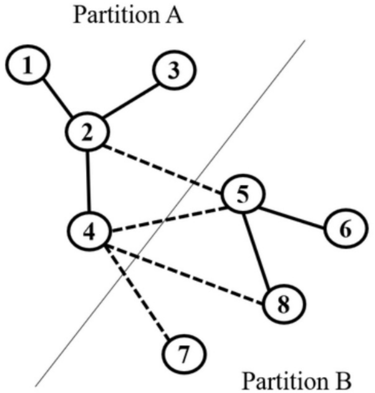
Task: identify unusual patterns of potential fraud occurrences

Classical Methods

Graph Partition

Kernighan-Lin 1970

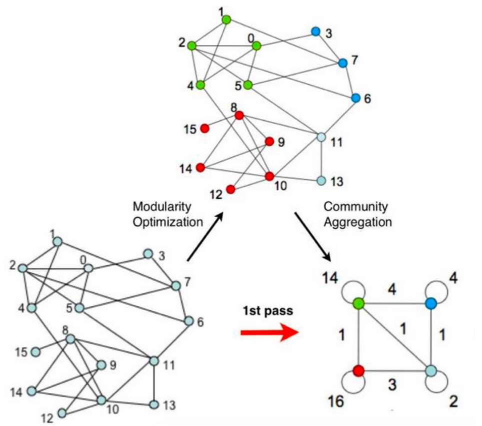
Barnes 1982



Optimisation

J. Stat. Mech. ’08

Phys. A ’16

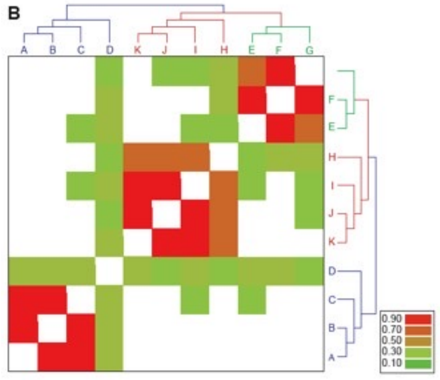


Hierarchical Clustering

PNAS’02

Phys. Rev. E ’04

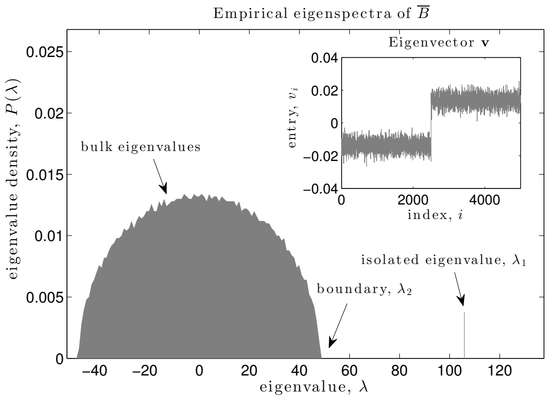
Phys. A ’18



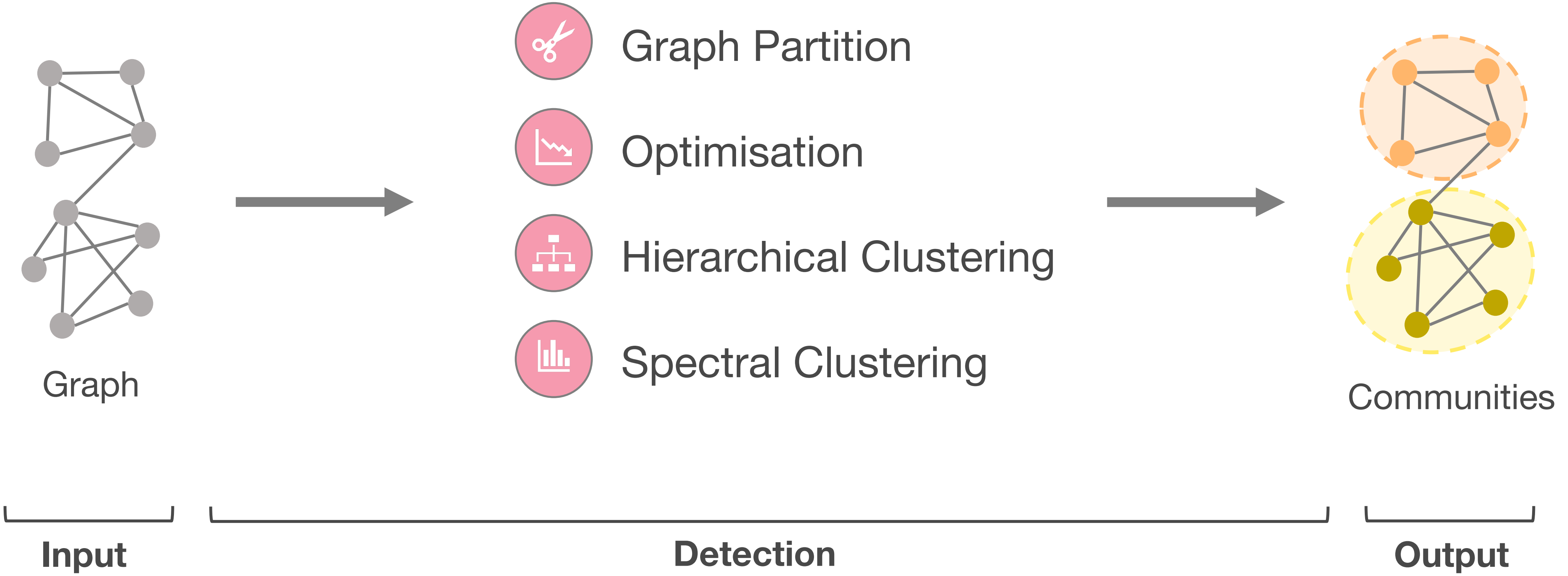
Spectral Clustering

Ann. Stat. ’13

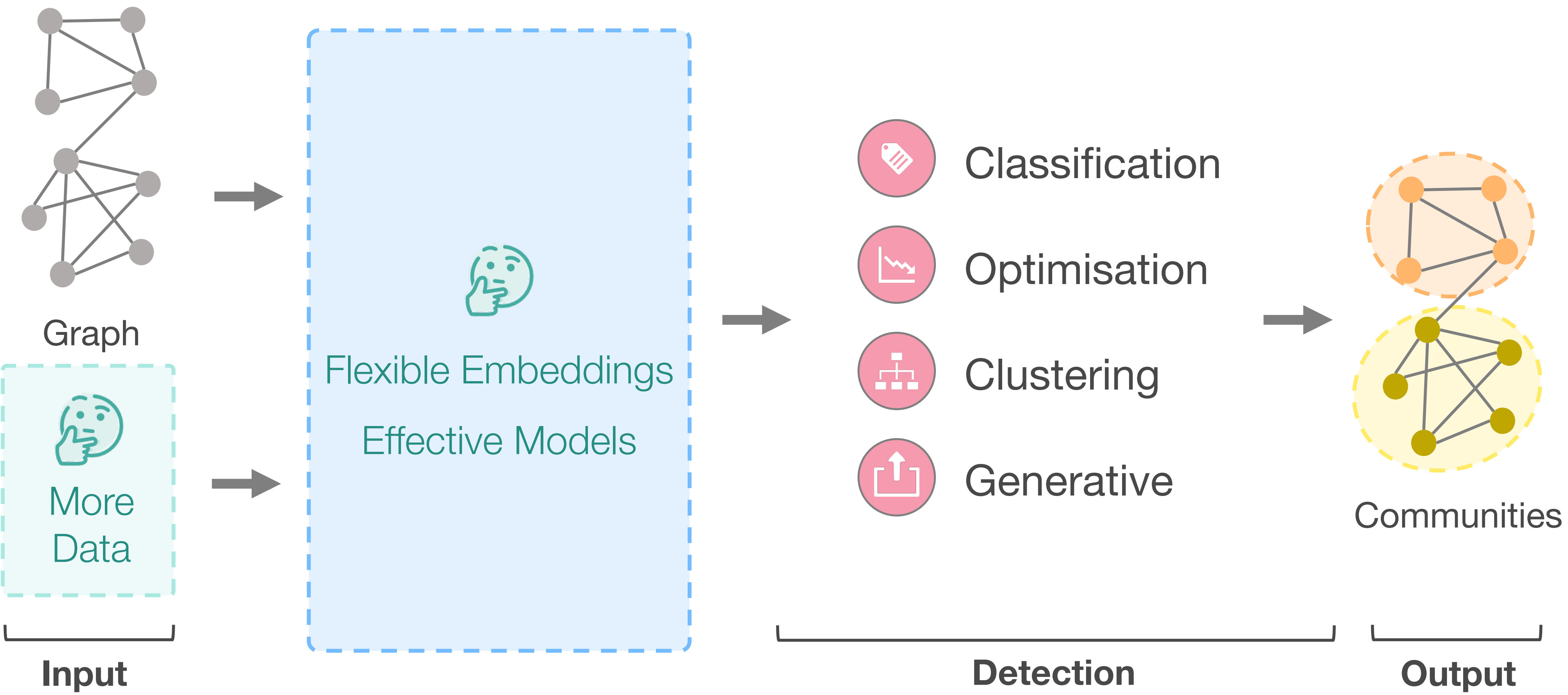
Comput. Neurosci. ’14



Classical Framework

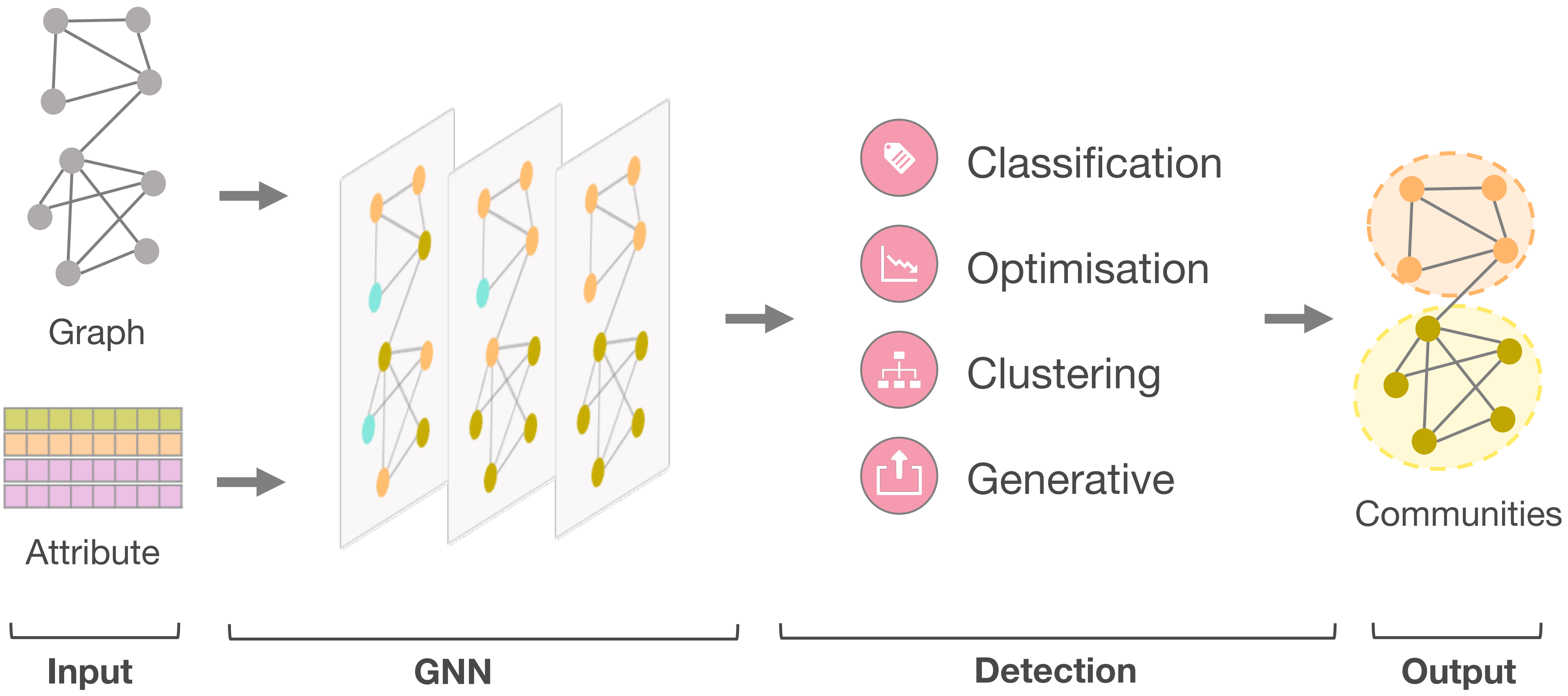


GNN CD Framework



Su, X., et al. (2022). A Comprehensive Survey on Community Detection with Deep Learning. IEEE Transactions on Neural Networks and Learning Systems.

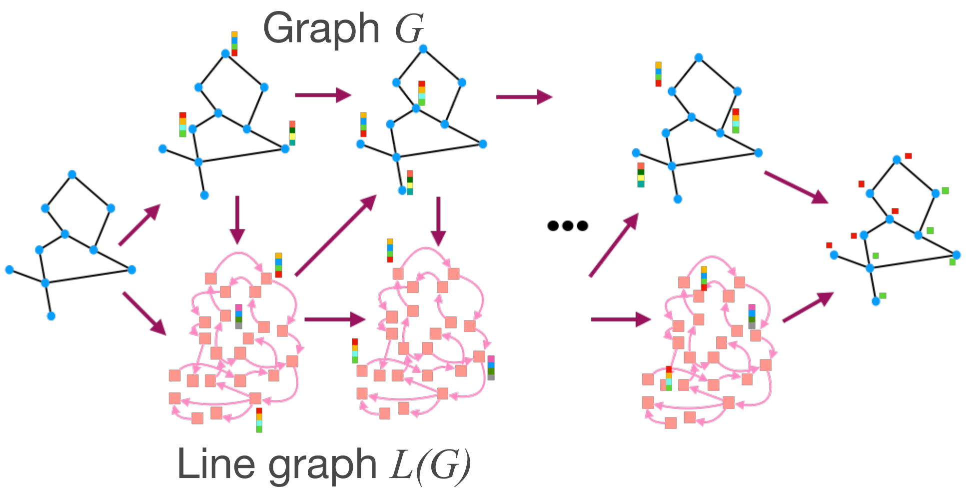
GNN CD Framework



Su, X., et al. (2022). A Comprehensive Survey on Community Detection with Deep Learning. IEEE Transactions on Neural Networks and Learning Systems.

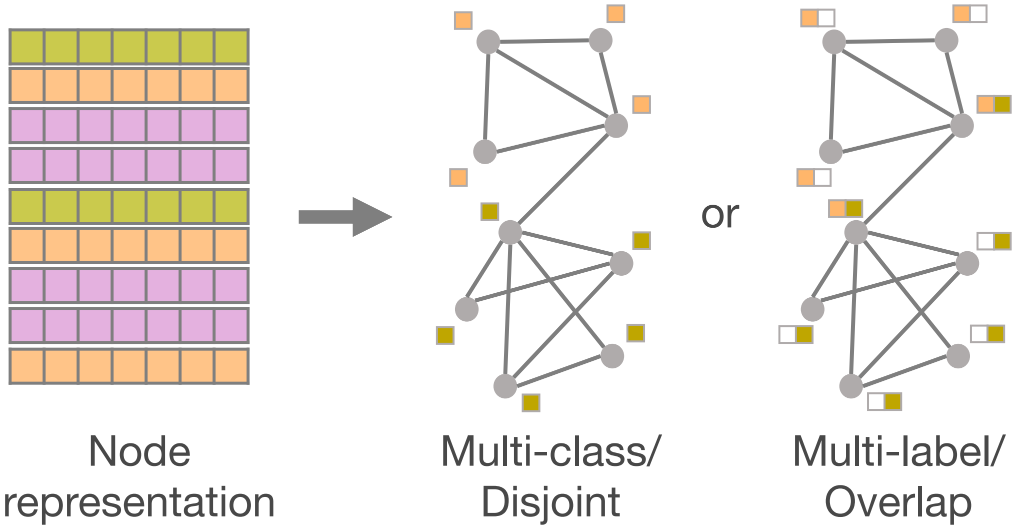
LGNN

GNN: Line GNN



- Simultaneous on graph and line graph
- Incorporate non-backtracking operator
- Represent edge adjacency information

Detection: Classification



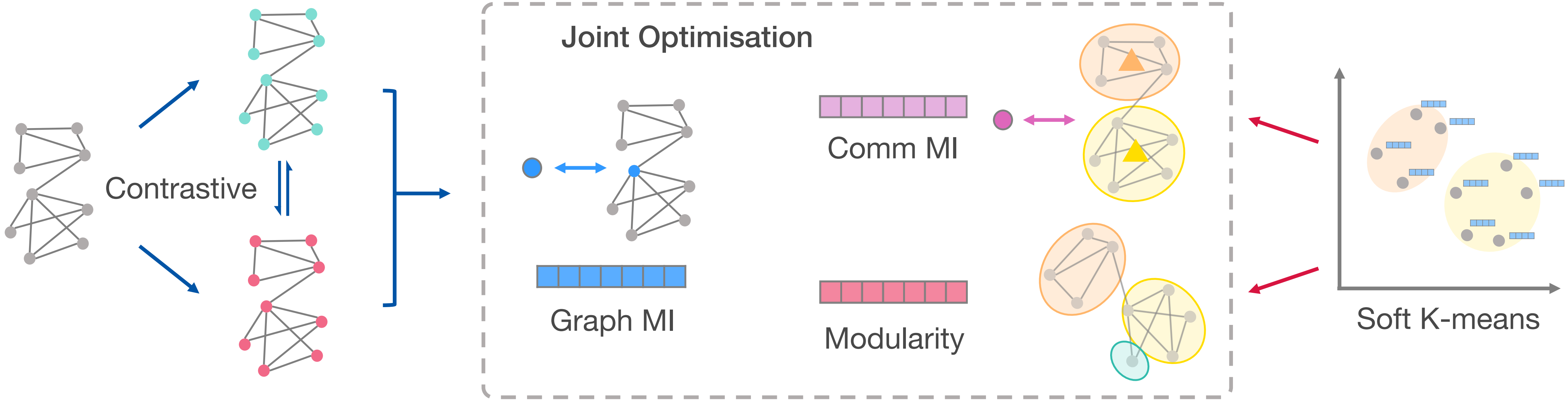
- Conventional GNN classification task
- Cross-entropy loss
- Require labelled data

Chen, Z., Li, X., & Bruna, J. (2019). Supervised Community Detection with Line Graph Neural Networks. 7th International Conference on Learning Representations. ICLR 2019.

CommDGI

GNN: Deep Graph Infomax

Detection: Joint Optimisation



- Maximise graph mutual information
- Contrastive method of negative samples
- Unsupervised MI objective

- Differentiable K-means clustering
- Soft K-means on representation
- Optimise community MI and modularity

Zhang, T., et al. (2020). CommDGI: Community Detection Oriented Deep Graph Infomax. Proceedings of the 29th ACM International Conference on Information & Knowledge Management, 1843–1852.

CD: Summary

- Learning-based methods such as GNNs improve the CD by more flexible model designs and data processing
- The GNN for CD framework usually includes a **GNN representation** module and a **detection** module

	LGNN	CommDGI
Paradigm	Supervised	Unsupervised
Community	Disjoint/Overlap	Disjoint
GNN	Line GNN	Deep Graph Infomax
Detection	Classification	Joint Optimisation

Outline

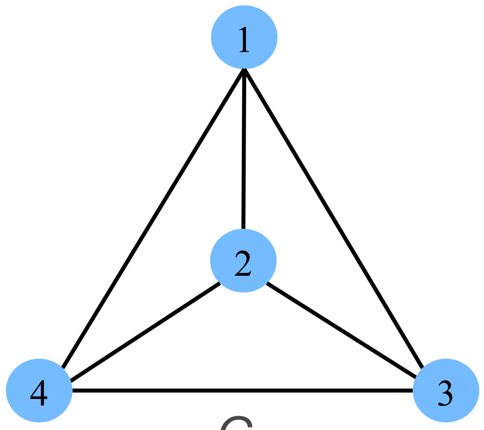
- Introduction
- Community Search
- Community Detection
- Q&A
- **Maximum Common Subgraph**
  Ningyi Liao 12 min
- Subgraph Isomorphism Counting
- Conclusion & Future Directions
- Q&A

Graph Isomorphism

Graph Isomorphism (non-labeled):

Given two graphs $G_1=(V_1, E_1)$ and $G_2=(V_2, E_2)$, there exists a bijection $f: V_1 \rightarrow V_2$ such that:

$$\text{edge } uv \in E_1 \iff \text{edge } f(u)f(v) \in E_2$$

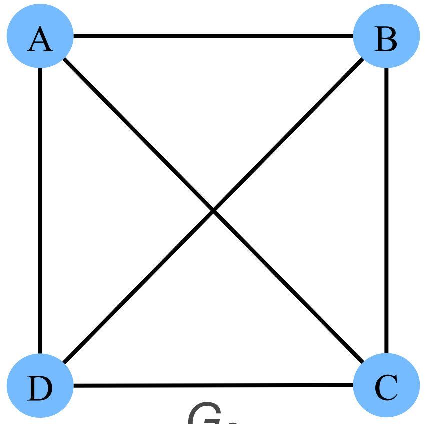


G_1

Bijection:

$f(1) = A$
$f(2) = B$
$f(3) = C$
$f(4) = D$

✓ isomorphic



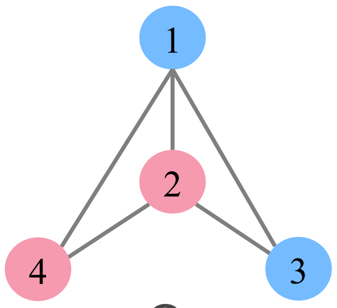
G_2

Graph Isomorphism

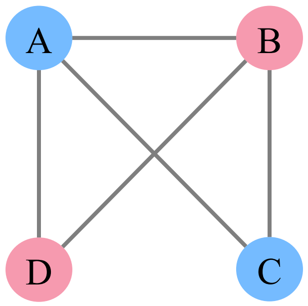
Graph Isomorphism (labeled):

Given two graphs $G_1=(V_1, E_1, L_1)$ and $G_2=(V_2, E_2 , L_2)$, there exists a bijection $f: V_1 \rightarrow V_2$, such that:

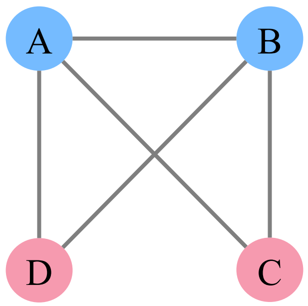
1) *Edge: $uv \in E_1 \;\Leftrightarrow\;$ edge $f(u)f(v) \in E_2$*
2) *Label: $L_1(v) = L_2(f(v))$*



G_1



✓ isomorphic



✗ not isomorphic

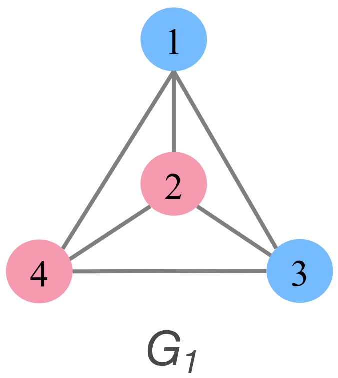
Max Common Subgraph

MCS: Max Common Subgraph (labeled, node-induced):

Given two graphs $G_1=(V_1, E_1, L_1)$ and $G_2=(V_2, E_2, L_2)$, find the largest sets $V_1' \subseteq V_1$ and $V_2' \subseteq V_2$, there exists a bijection $f: V_1' \rightarrow V_2'$, such that:

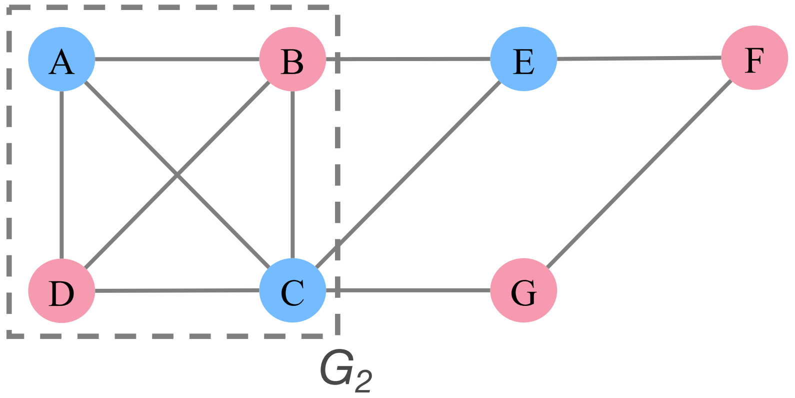
1) $u, v \in V_1'$, *edge* $uv \in E_1 \Leftrightarrow$ *edge* $f(u)f(v) \in E_2$

2) $v \in V_1'$, *vertex label* $L_1(v) = L_2(f(v))$



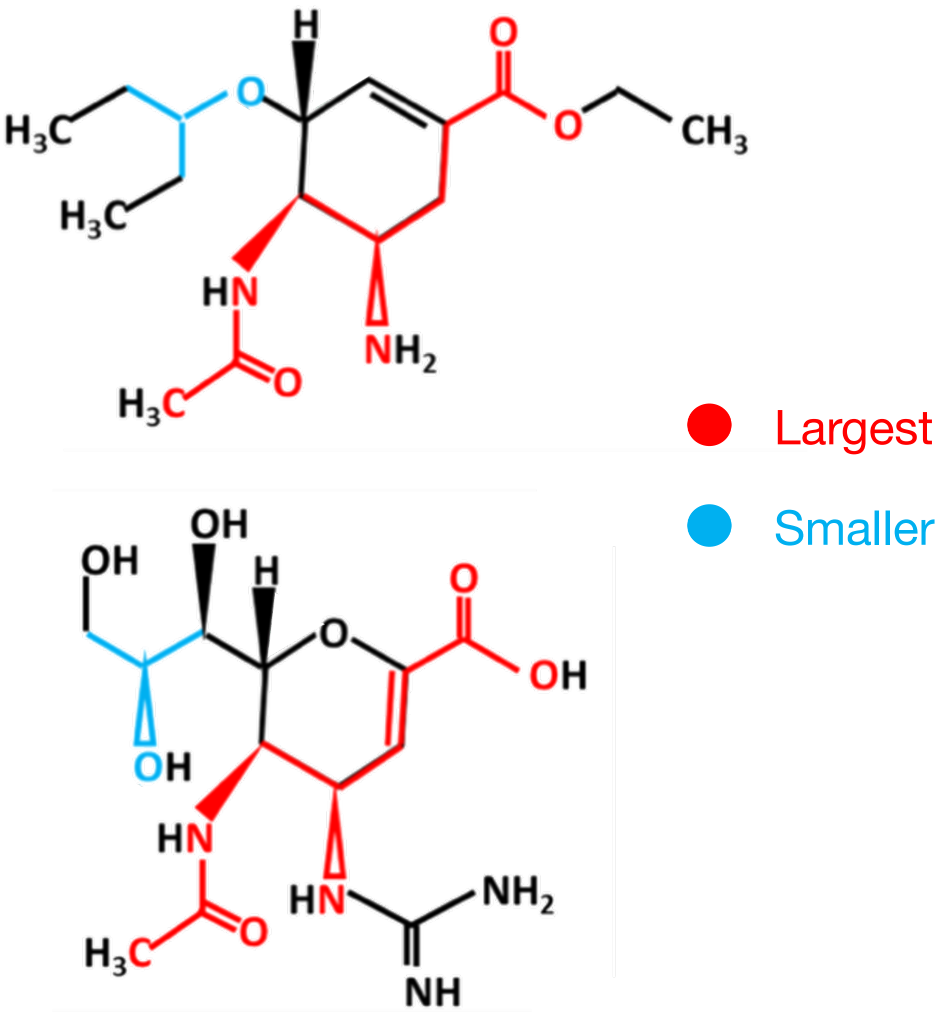
Bijection:

$f(1) = A$
$f(2) = B$
$f(3) = C$
$f(4) = D$



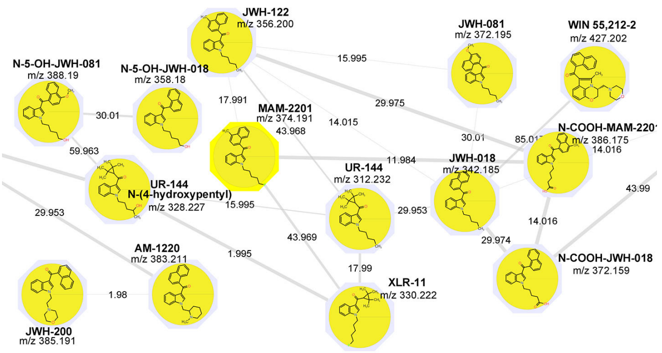
MCS: The Applications

- **Graph:** molecule
- **Vertex:** atom
- **Edge:** chemical bond
- **Task:** Molecules that have similar partial structures are expected to have similar drug efficacy. *How to find the maximum common partial structures in two molecules?*

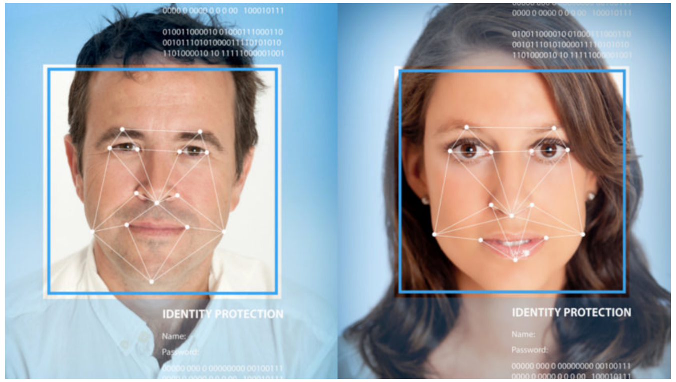


Yasuharu Okamoto. 2020. Finding a Maximum Common Subgraph from Molecular Structural Formulas through the Maximum Clique Approach Combined with the Ising Model. ACS Omega 5 (22), 13064-13068.

MCS: The Applications




```
NtCreateDirectoryObject(OUT DirectoryHandle -> 1,
                        … ,  IN ObjectAttributes -> A);
.....
NtCreateFile(OUT FileHandle -> 2, ... ,
             IN ObjectAttributes -> B,..........);
.....
NtCreateFile(OUT FileHandle -> 3, ... ,
             IN ObjectAttributes -> C,..........);
.....
NtCreateSection(OUT SectionHandle -> 4, ... ,
                IN ObjectAttributes->D, ....,
                IN FileHandle -> 2);
```



Molecule Search

Graph: molecule graph DB

Task: find molecules in DB similar to query graph

Software Analysis

Vertex: kernel object | Edge: call

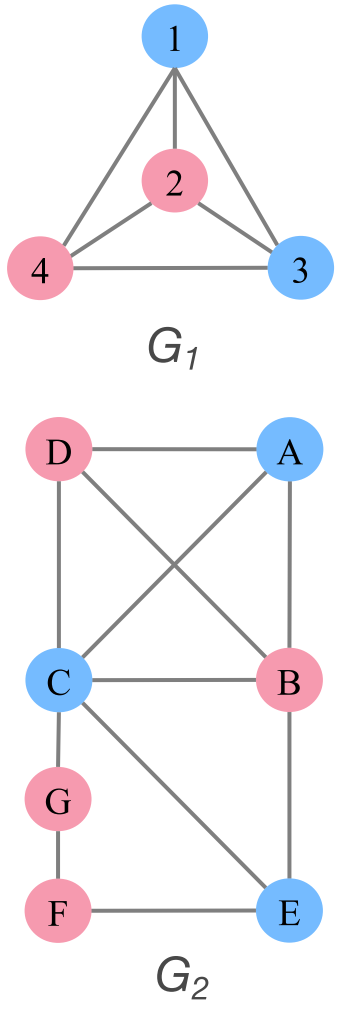
Task: discover specific malware behaviours in software

Facial Recognition

Vertex: landmark in image

Task: compare similarity of given image to DB

Conventional Solution: Branch and Bound

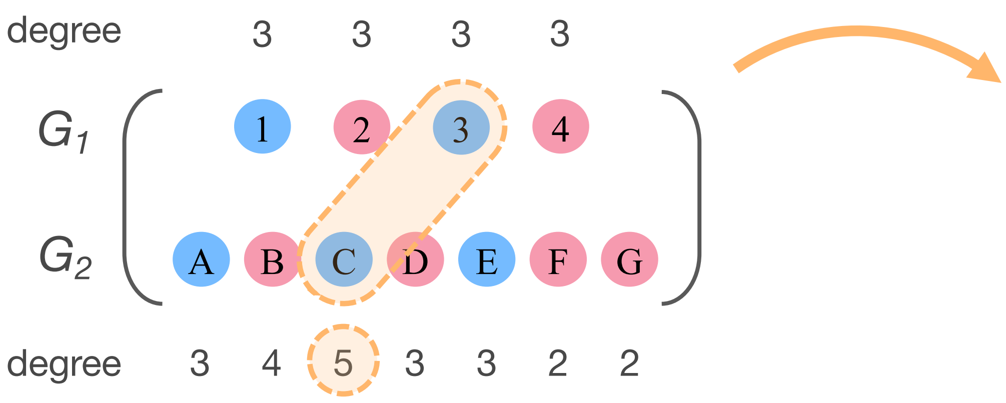


Candidate Vertices

Connectivity

Branch

Max degree



G_1'

G_2'

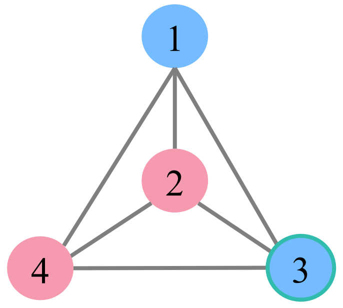
Bound

Potential Max Size

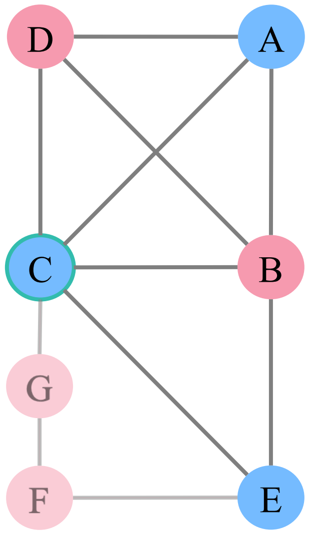
> Current Max Size

C. McCreesh, P. Prosser, and J. Trimble. 2017. A Partitioning Algorithm for Maximum Common Subgraph Problems. In 26th IJCAI. 712–719.

Conventional Solution: Branch and Bound



G_1



G_2

Candidate Vertices: Connectivity

Branch: Max degree

G_1': 3

G_2': C

Bound

Potential Max Size > Current Max Size

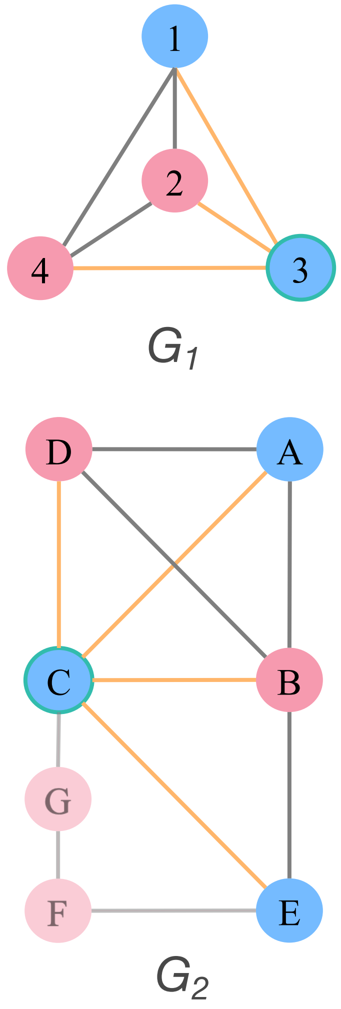
4 > 1

✓ *Continue*

degree	3	3	3				
G_1	1	2	4				
G_2	A	B	D	E	|	F	G
degree	3	4	3	3		2	2

C. McCreesh, P. Prosser, and J. Trimble. 2017. A Partitioning Algorithm for Maximum Common Subgraph Problems. In 26th IJCAI. 712–719.

Conventional Solution: Branch and Bound



Candidate Vertices

Connectivity

Branch

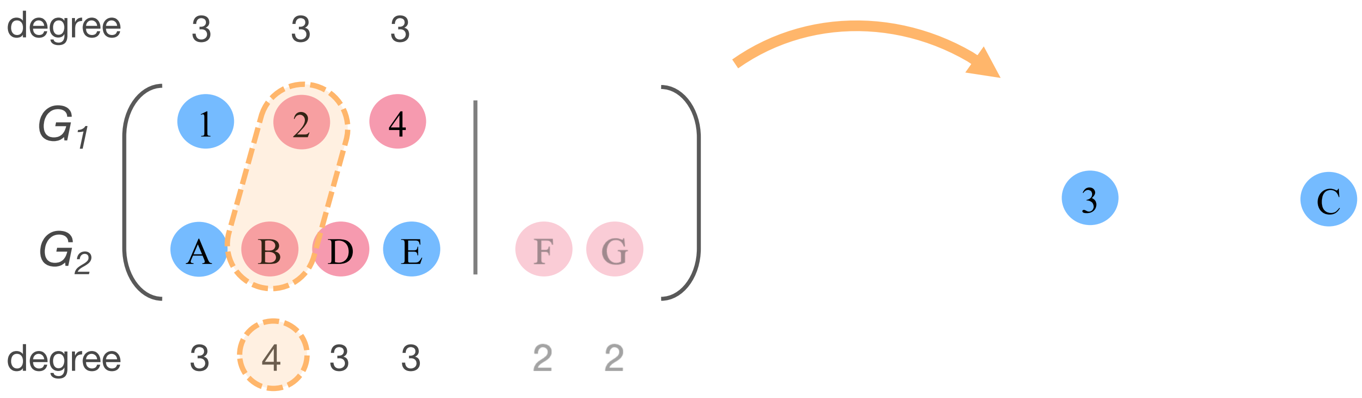
Max degree

G_1' G_2'

Bound

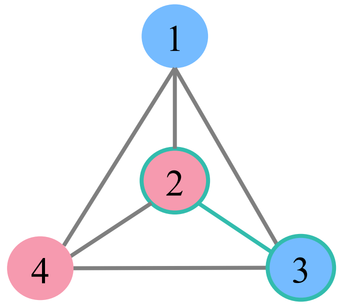
Potential Max Size

> Current Max Size

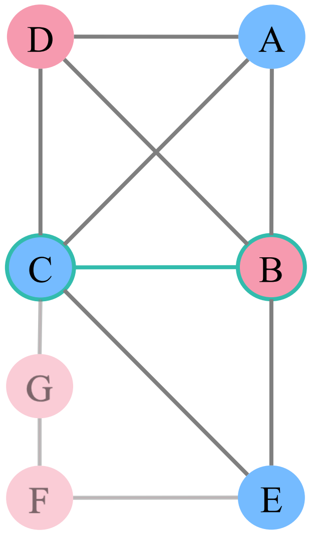


C. McCreesh, P. Prosser, and J. Trimble. 2017. A Partitioning Algorithm for Maximum Common Subgraph Problems. In 26th IJCAI. 712–719.

Conventional Solution: Branch and Bound



G_1



G_2

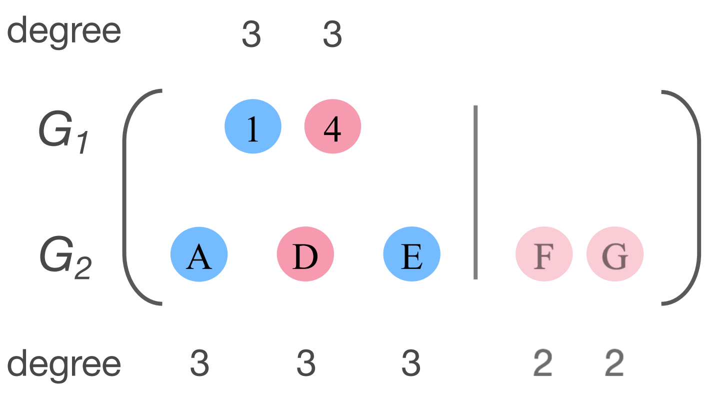
Candidate Vertices | **Branch** | G_1' | G_2'

Connectivity | Max degree

Bound

Potential Max Size

> Current Max Size

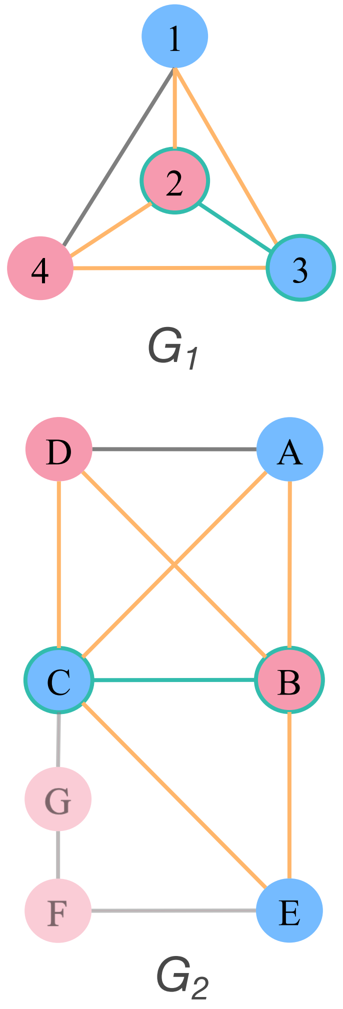


4 > 2

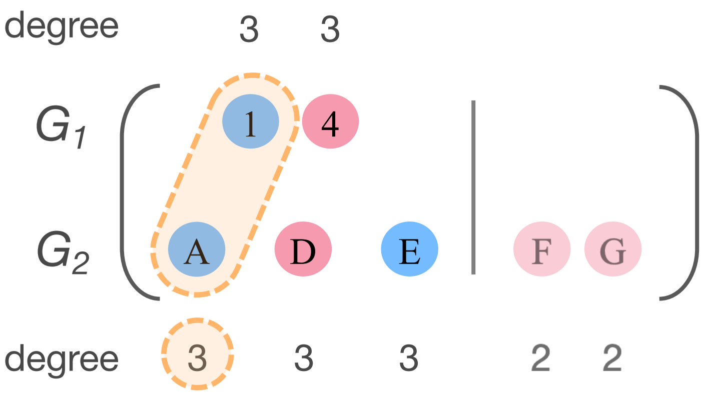
✓ *Continue*

C. McCreesh, P. Prosser, and J. Trimble. 2017. A Partitioning Algorithm for Maximum Common Subgraph Problems. In 26th IJCAI. 712–719.

Conventional Solution: Branch and Bound

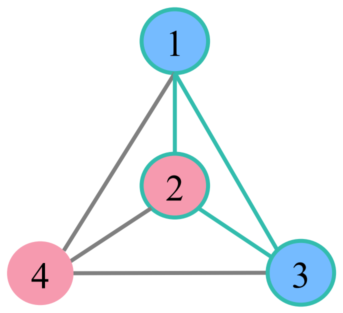


Candidate Vertices	Branch	G_1'	G_2'	Bound
Connectivity	Max degree			Potential Max Size > Current Max Size

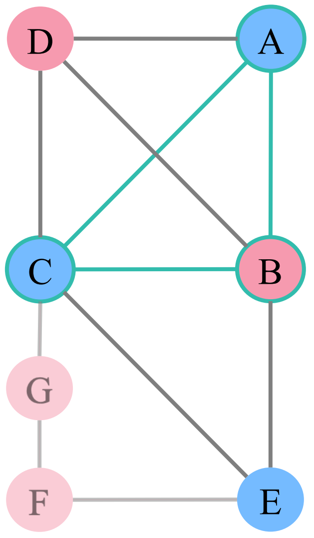


C. McCreesh, P. Prosser, and J. Trimble. 2017. A Partitioning Algorithm for Maximum Common Subgraph Problems. In 26th IJCAI. 712–719.

Conventional Solution: Branch and Bound



G_1



G_2

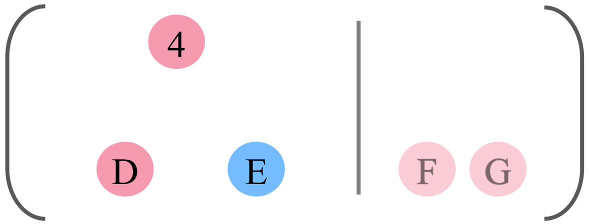
Candidate Vertices | **Branch** | G_1' | G_2'

Connectivity | Max degree

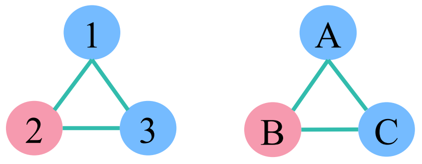
degree 3

G_1

G_2



degree 3 3 2 2



Bound

Potential Max Size

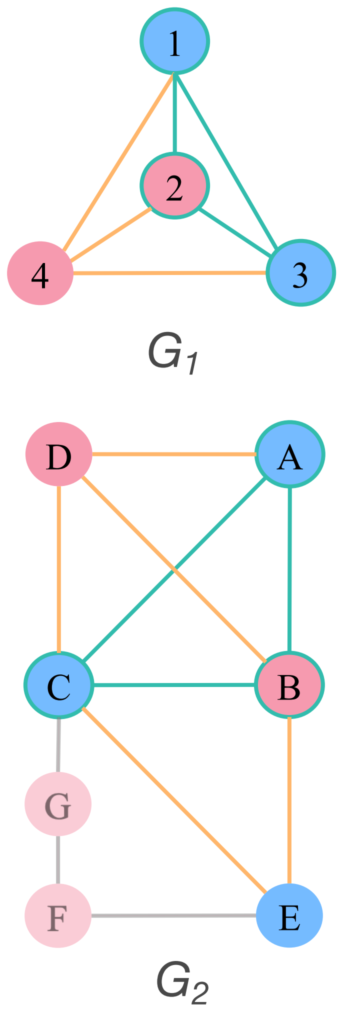
> Current Max Size

4 > 3

✓ *Continue*

C. McCreesh, P. Prosser, and J. Trimble. 2017. A Partitioning Algorithm for Maximum Common Subgraph Problems. In 26th IJCAI. 712–719.

Conventional Solution: Branch and Bound



Candidate Vertices

Connectivity

Branch

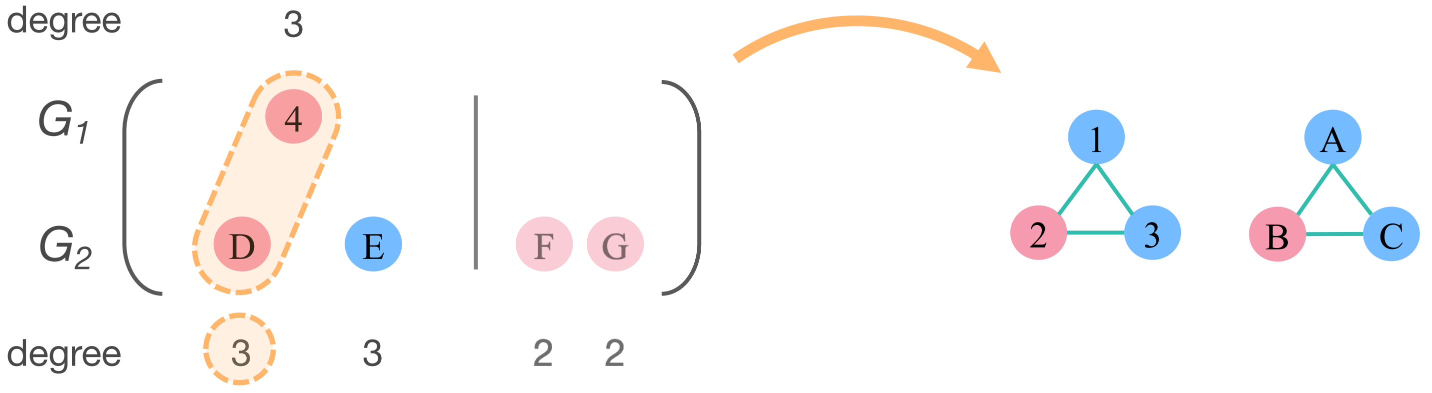
Max degree

G_1' G_2'

Bound

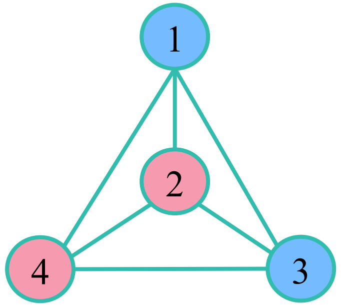
Potential Max Size

> Current Max Size

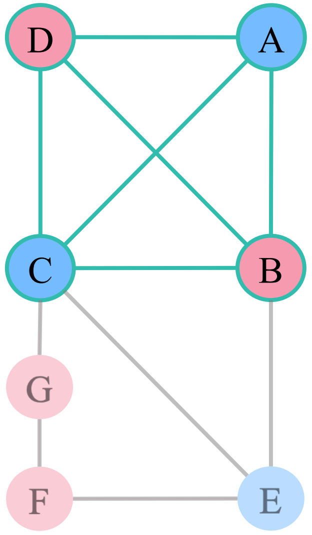


C. McCreesh, P. Prosser, and J. Trimble. 2017. A Partitioning Algorithm for Maximum Common Subgraph Problems. In 26th IJCAI. 712–719.

Conventional Solution: Branch and Bound



G_1



G_2

Candidate Vertices	**Branch**	G_1'	G_2'
Connectivity	Max degree		

Bound

Potential Max Size

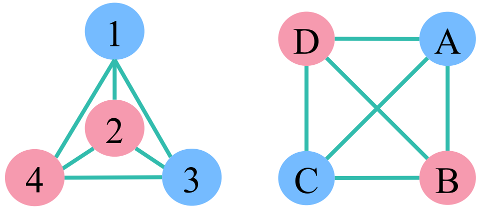
> Current Max Size

degree

G_1

G_2 E | F G

degree 3 | 2 2



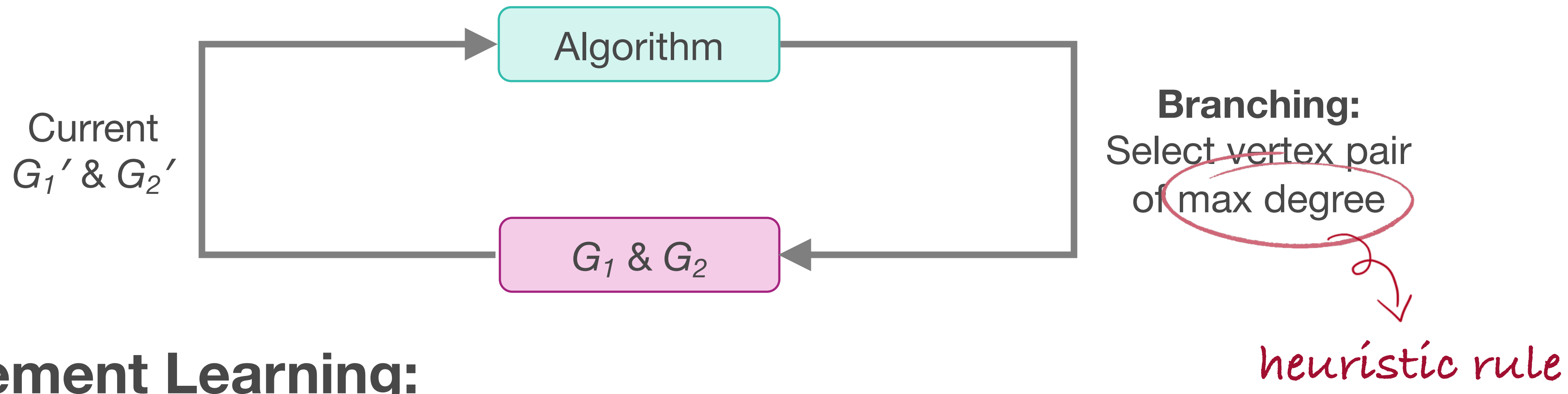
✓ *current best*

Current Max Size = 4

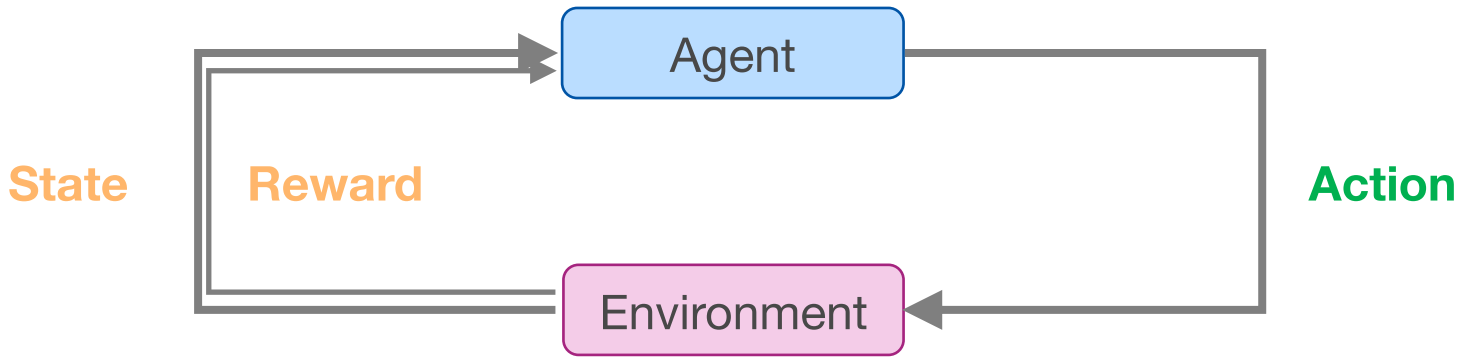
C. McCreesh, P. Prosser, and J. Trimble. 2017. A Partitioning Algorithm for Maximum Common Subgraph Problems. In 26th IJCAI. 712–719.

MCS Search Framework

McSplit Branch and Bound:

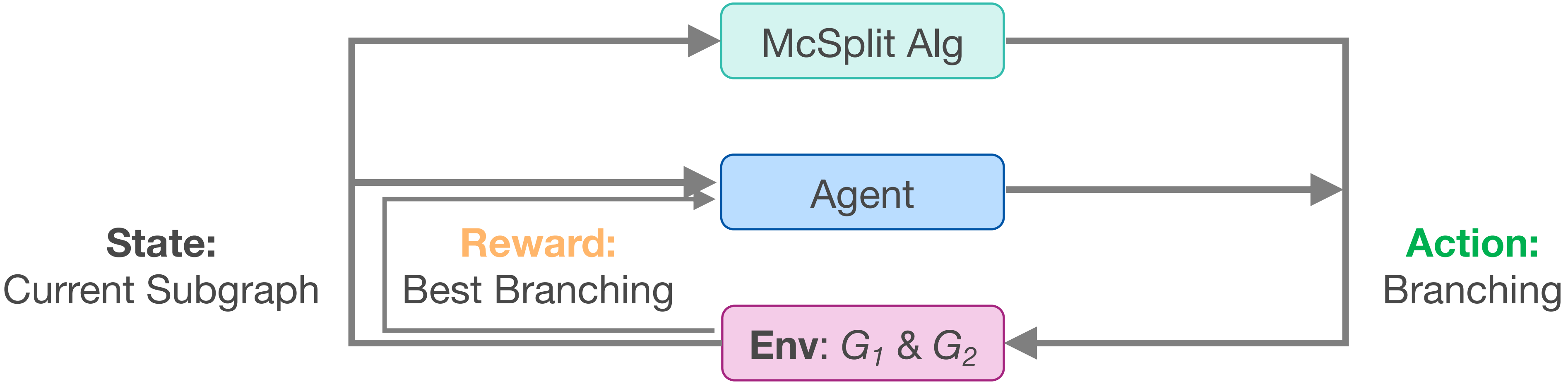


Reinforcement Learning:



McSplit+RL

RL alongside BnB Search:



Reward Design

Reach search tree leaf as early as possible

⇕

Minimise the size of the search space

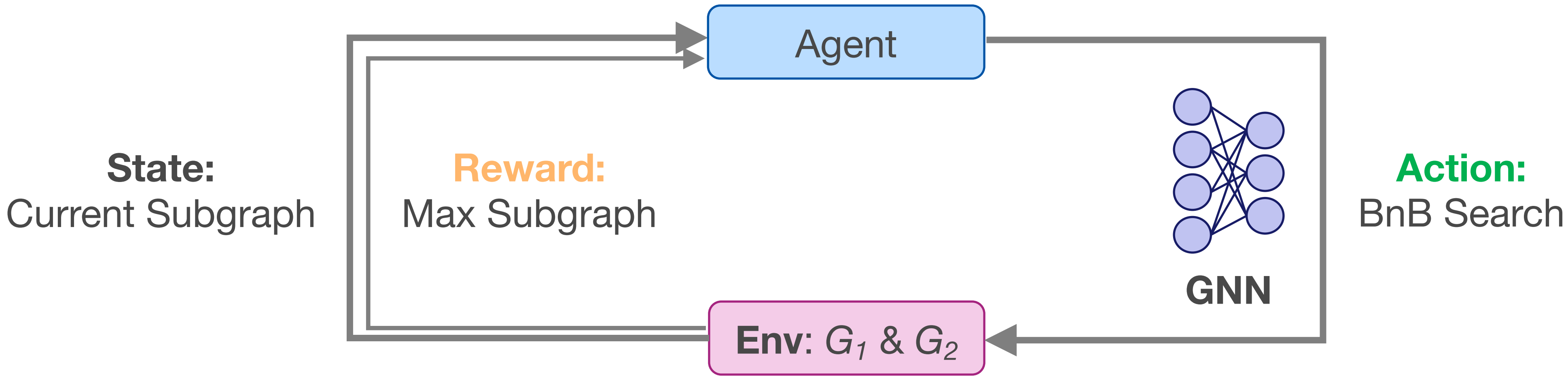
Action Design

McSplit	Vertex pair with largest degree
	⇓
McSplit+RL	Vertex pair with best RL reward

Y. Liu, C. M. Li, H. Jiang, and K. He. 2020. A Learning Based Branch and Bound for Maximum Common Subgraph Related Problems. Proceedings of the AAAI Conference on Artificial Intelligence 34, 03 (2020), 2392–2399.

GLSearch

End-to-end RL BnB Search:



Reward Design

Reward of RL: find the max subgraph

Goal of GNN: learn to achieve best reward

Action Design

GNN learning the current state

Decide search order & select vertex pair

Y. Bai, D. Xu, Y. Sun, and W. Wang. 2020. GLSearch: Maximum Common Subgraph Detection via Learning to Search. In ICML. 588–598.

MCS: Summary

- The MCS problem is NP-hard. Conventional algorithms are based on Branch and Bound search under heuristic rules
- The search can be powered by **Reinforcement Learning**: design reward (learning goal) and action (one step of search)
- RL can improve the search by reaching solutions faster

Model	Reward	Action
McSplit+RL	Optimise BnB search	Select vertex of best reward
GLSearch	Find max subgraph	Perform BnB search

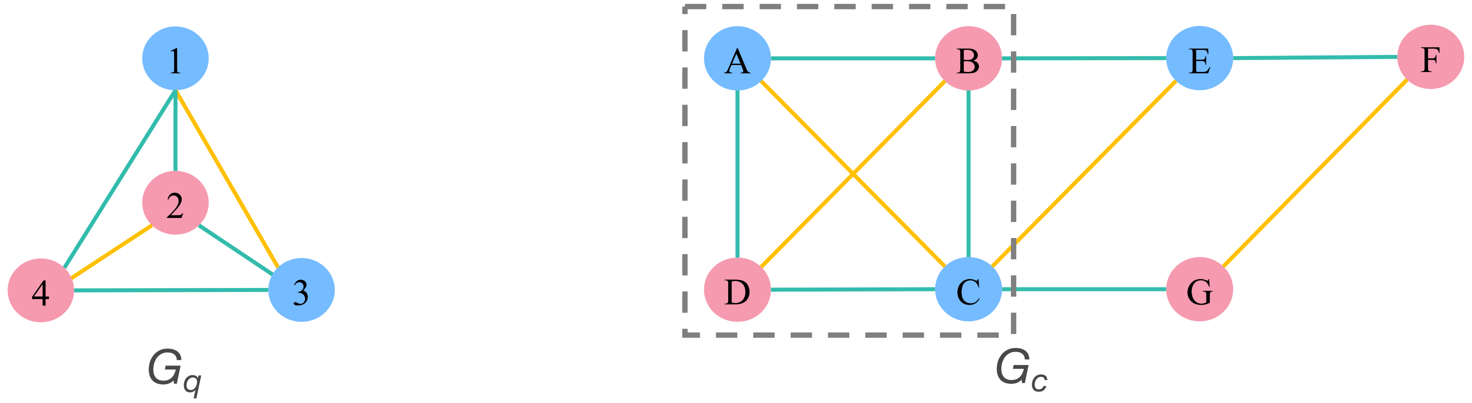
Outline

- Introduction
- COMMUNITY SEARCH
- COMMUNITY DETECTION
- Q&A
- MAXIMUM COMMON SUBGRAPH
- SUBGRAPH ISOMORPHISM COUNTING — Reynold Cheng, 12 min
- Conclusion & Future Directions
- Q&A

Subgraph Isomorphism Counting

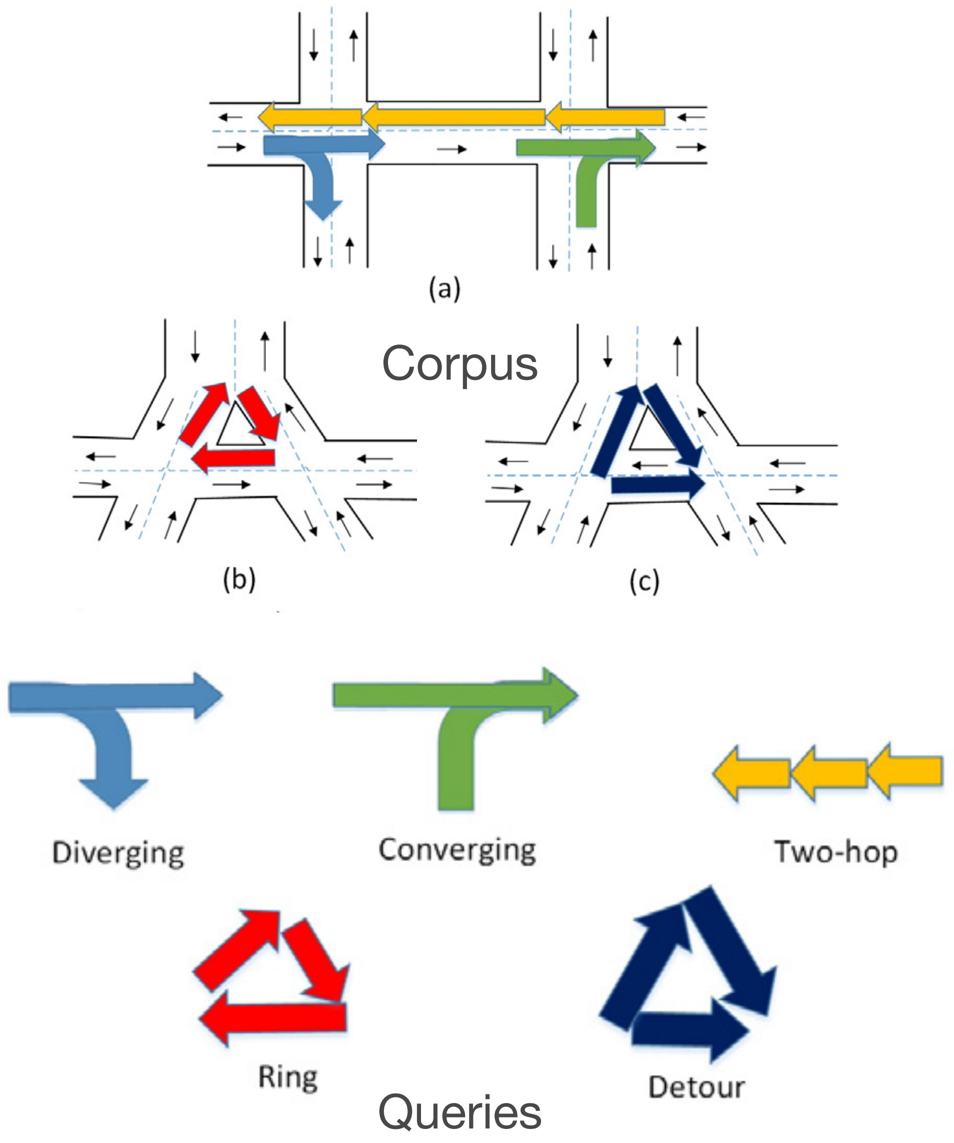
SIC: Subgraph Isomorphism Counting (labeled, heterogeneous):

Given a query graph $G_q=(V_q, E_q, L_q, C_q)$ *and a corpus graph* $G_c=(V_c, E_c, L_c, C_c)$*, return the number of subgraphs in* G_c *such that those subgraphs are isomorphic to* G_q



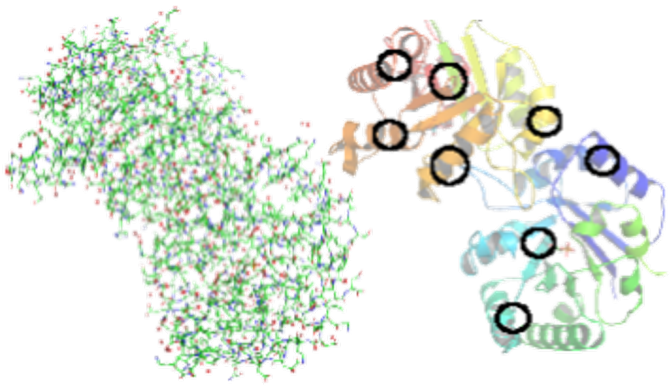
SIC: The Applications

- **Corpus Graph:** road network
- **Query Graph:** connectivity patterns
- **Vertex:** intersections
- **Edge:** road segments
- **Task:** *What is the frequency of certain connectivity patterns in a road network?*



G Shen, et al. 2022. Motif discovery based traffic pattern mining in attributed road networks. Knowledge-Based Systems 250.

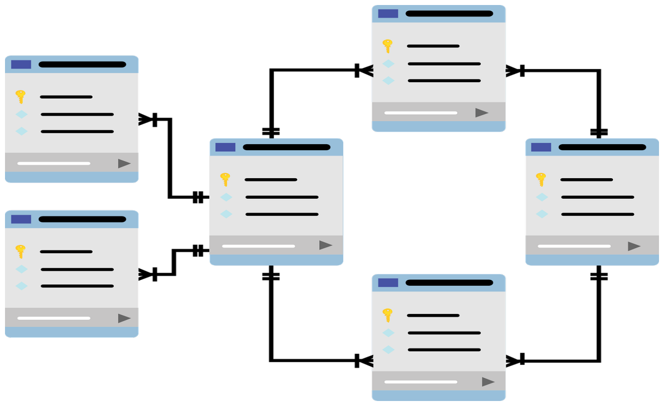
SIC: The Applications



Protein Structure

Graph: protein interaction

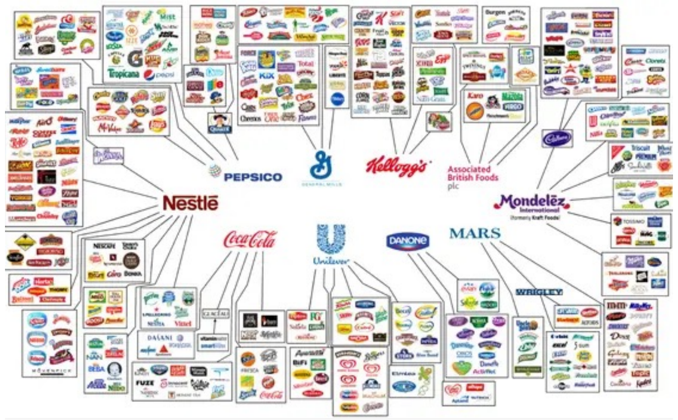
Task: count frequency of certain interaction patterns in DB



DBMS Bug Detection

Graph: DBMS schema tables

Task: find redundant queries in the schema graph



Pattern Discovery

Graph: inter-firm network

Task: identify and count certain connection patterns

SIC: The Challenge

- Exact SIC problem is NP-hard, resulting in exponential complexity

Brute force search	Ullmann (1976)	VF2 (2007)
$O\left(\frac{n_c!}{(n_c - n_q)!} \cdot d^{n_q}\right)$	$O\left(n_q^{n_c} \cdot n_c^2\right)$	$O(n_c^2) \sim O(n_c!)$

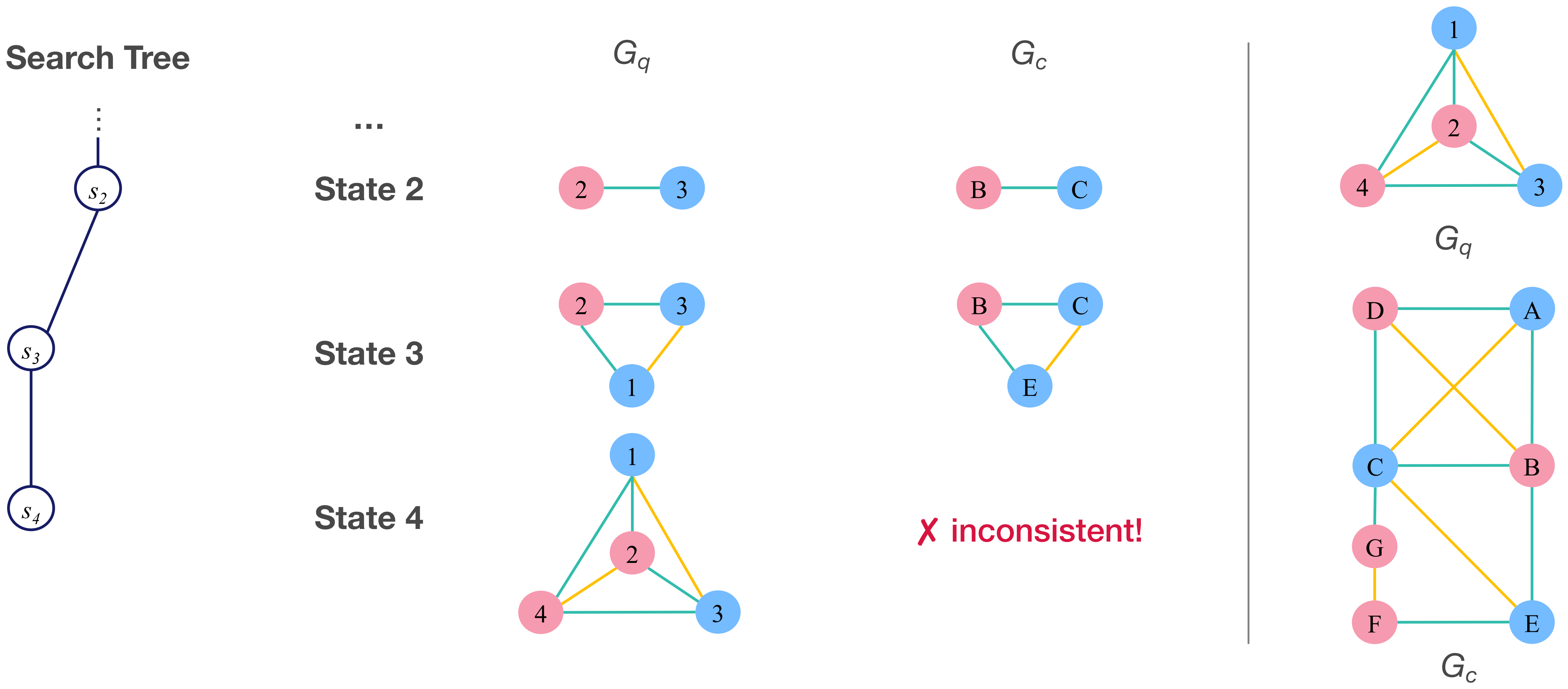
Corpus Nodes Size

Avg Degree

Query Nodes Size

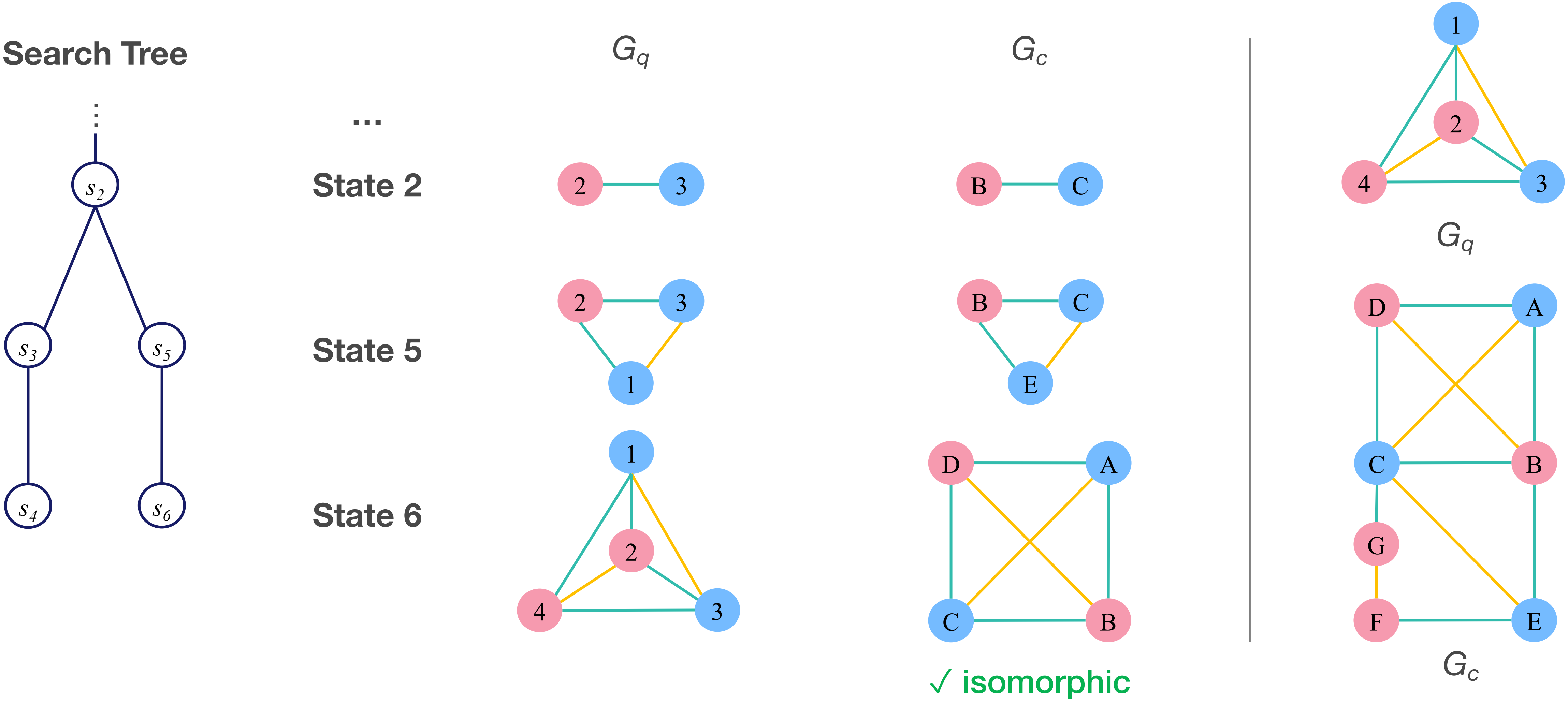
Ribeiro, P., Paredes, P., Silva, M. E. P., Aparicio, D., & Silva, F. (2022). A Survey on Subgraph Counting: Concepts, Algorithms, and Applications to Network Motifs and Graphlets. ACM Computing Surveys, 54(2), 1–36.

Conventional Solution: Tree Search



Luigi P. Cordella, Pasquale Foggia, Carlo Sansone, and Mario Vento. 2004. A (Sub)Graph Isomorphism Algorithm for Matching Large Graphs. PAMI 26, 10 (2004), 1367–1372.

Conventional Solution: Tree Search

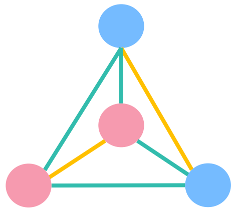


Luigi P. Cordella, Pasquale Foggia, Carlo Sansone, and Mario Vento. 2004. A (Sub)Graph Isomorphism Algorithm for Matching Large Graphs. PAMI 26, 10 (2004), 1367–1372.

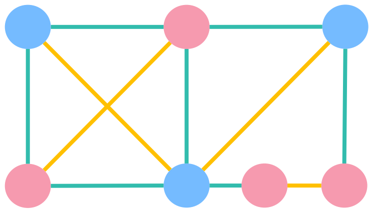
Question-Answering Framework

The SIC Problem:

Input



Query Graph



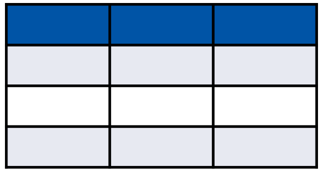
Corpus Graph

Output $C = 1$

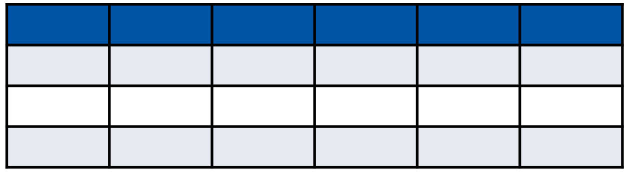
Answer of query

The Question-Answering Problem:

Input



Query



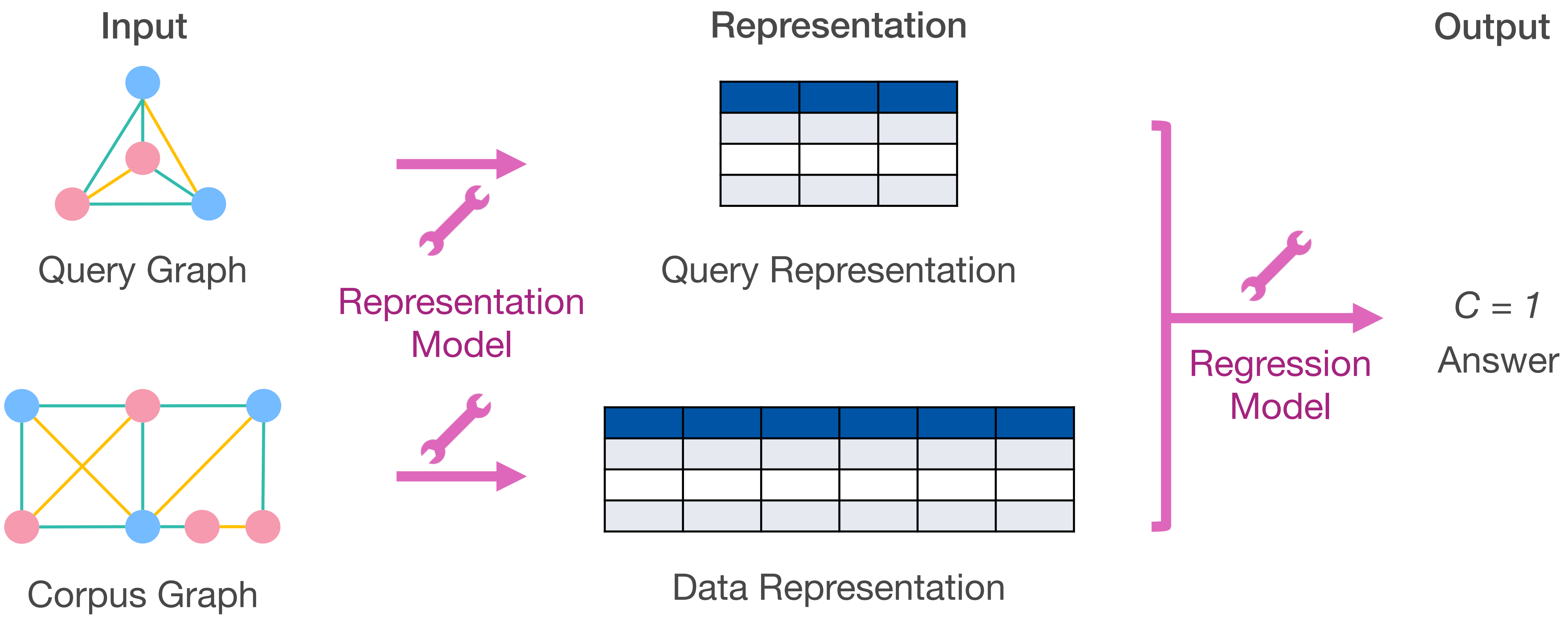
Data

Output $C = 1$

Answer of query

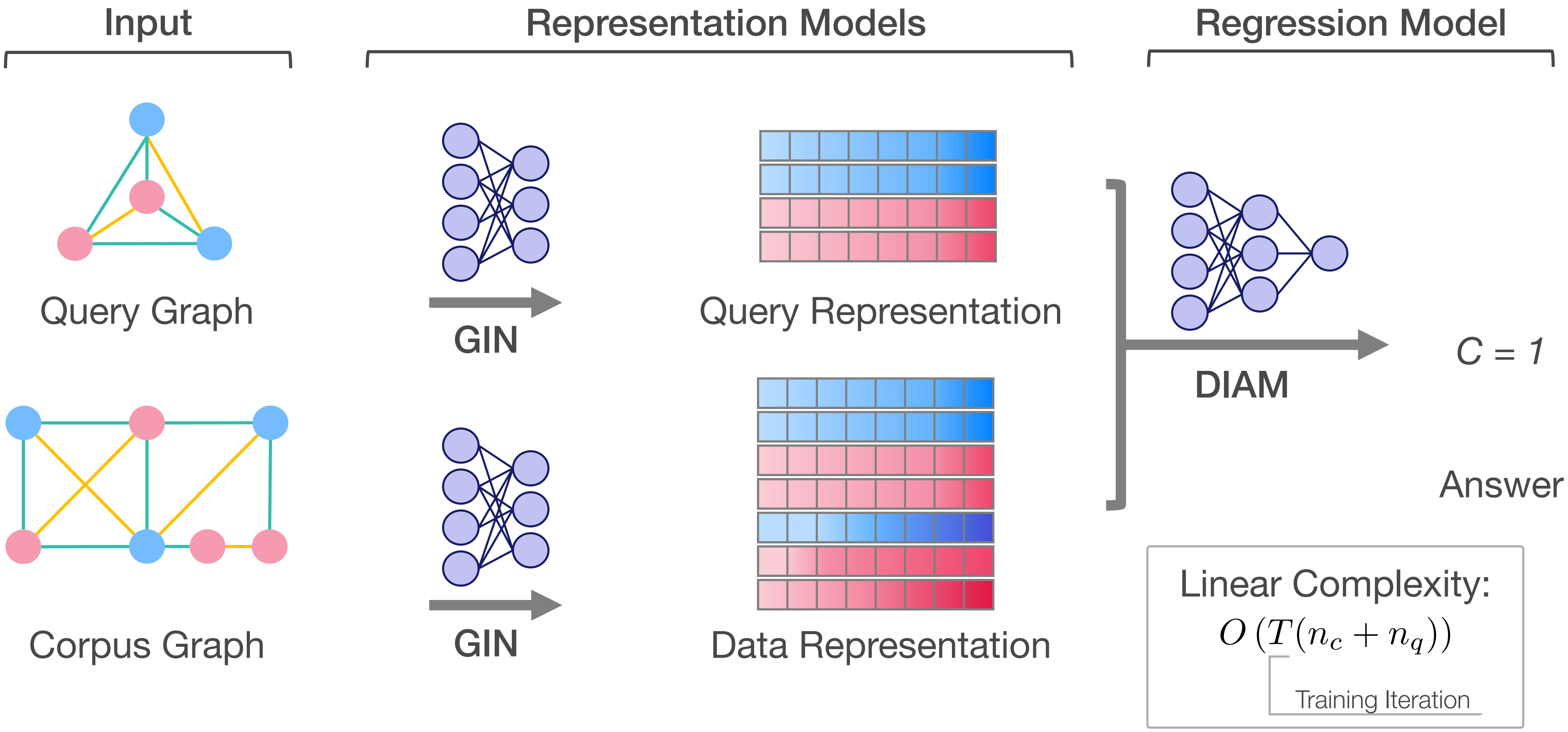
Question-Answering Framework

SIC Problem under QA Framework:



DIAMNet

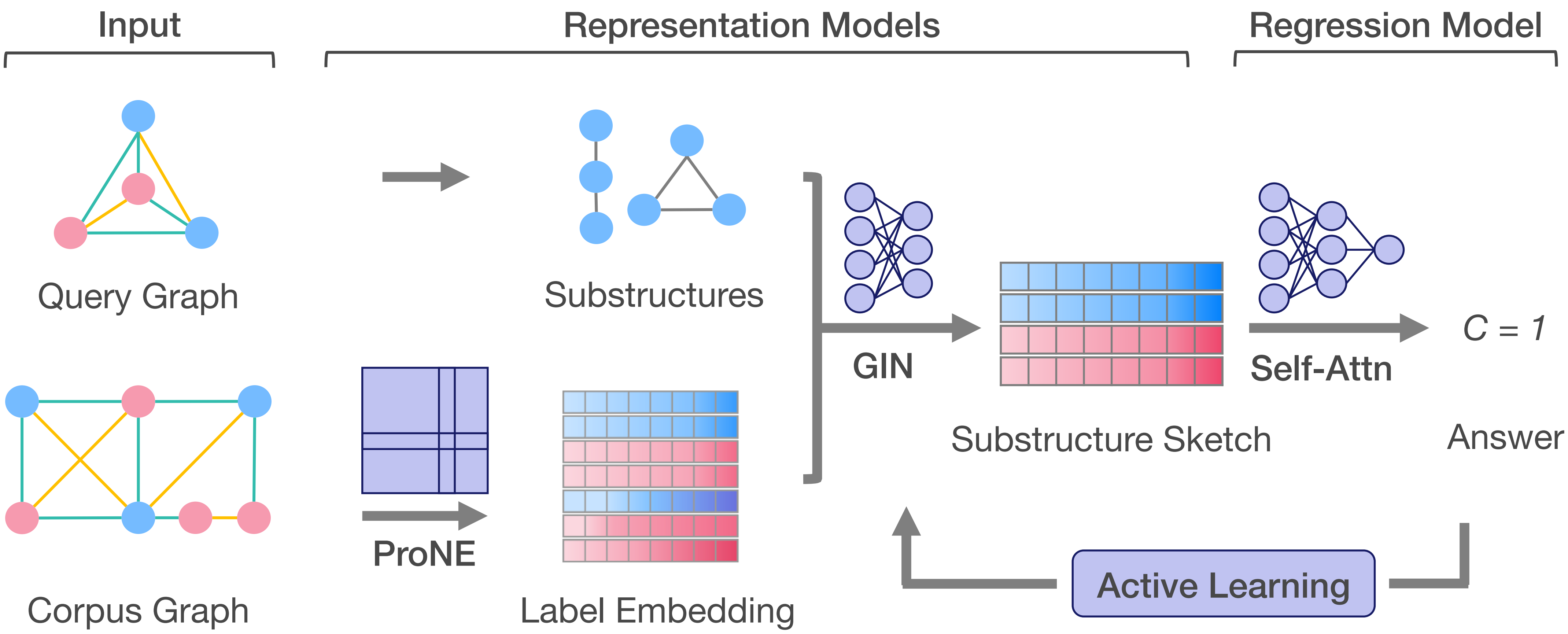
What are effective representation and regression models? <u>GIN + DIAM</u>



X. Liu, H. Pan, M. He, Y.u Song, X. Jiang, and L. Shang. 2020.
Neural Subgraph Isomorphism Counting. In SIGKDD. 1959–1969.

ALSS

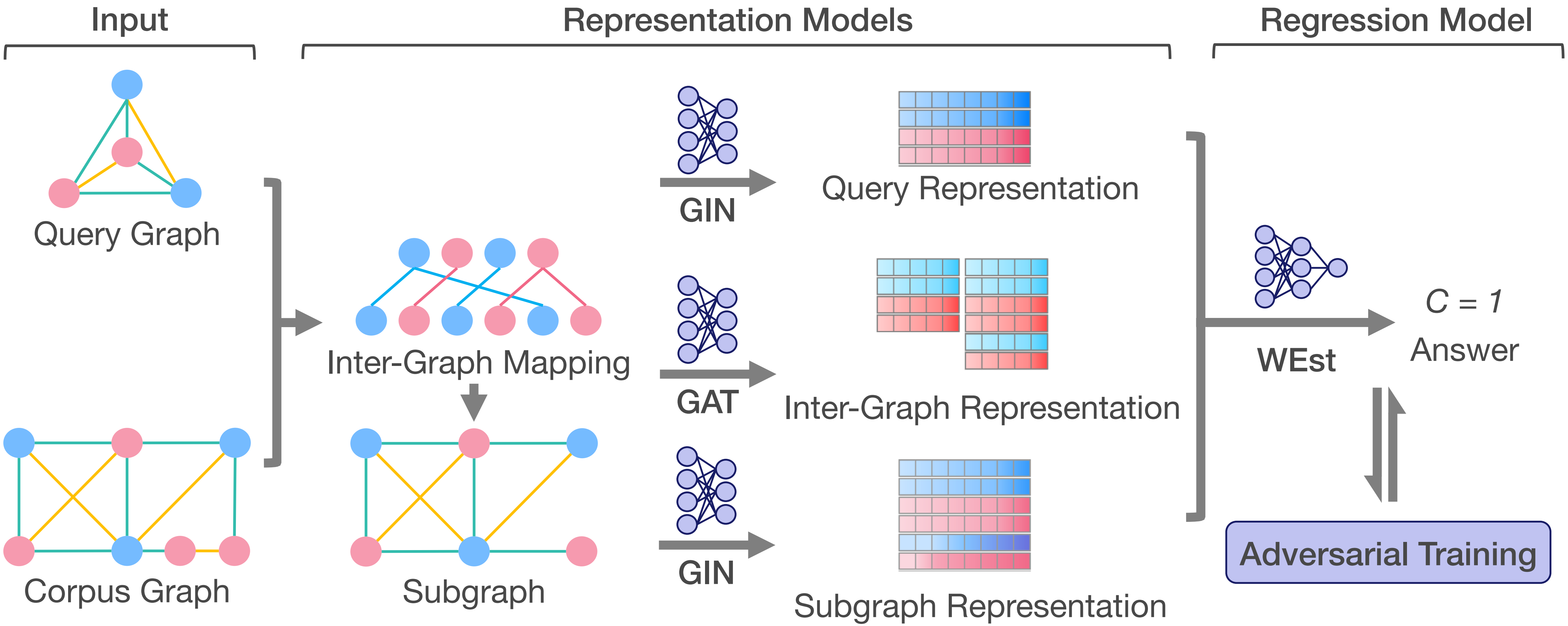
How to apply RDBMS techniques? Sketch Learning + Active Learning



K. Zhao, J. X. Yu, H. Zhang, Q. Li, and Y. Rong. 2021. A Learned Sketch for Subgraph Counting. In SIGMOD. 2142–2155.

NeurSC

How to apply learning-based techniques? <u>Inter-Graph + Adversarial Training</u>



H. Wang, R. Hu, Y. Zhang, L. Qin, W. Wang, and W. Zhang. 2022. Neural Subgraph Counting with Wasserstein Estimator. In SIGMOD. 160–175.

SIC: Summary

- The SIC problem is NP-hard. Conventional enumeration-based algorithm is limited by the graph size
- The **Question-Answering Framework** enables ML algorithms: representation (graph to embedding) & regression (estimate count)
- ML approaches output favourable estimation with linear complexity

Model	Representation	Regression
DIAMNet	GIN	Attention
ALSS	Sketch learning	Active learning
NeurSC	Intra- & inter-graph	Adversarial training

Outline

Introduction

Community Search

Community Detection

Q&A

Maximum Common Subgraph

Subgraph Isomorphism Counting

Conclusion & Future Directions Reynold Cheng, 10 min

Q&A

Summary: ML for Subgraph

Subgraph Problem	Paradigm	Algorithm	Advance	Method
COMMUNITY SEARCH	●	GNN	◎ ⏱	ICS-GNN [VLDB, 2021]
	●	GNN	◎ ⏱	QD-GNN [VLDB, 2022]
	●	GNN	◎ ⏱	CGNP [arxiv, 2022]
	◐	GNN	◎ ⏱ ⛶	COCLEP [ICDE, 2023]
COMMUNITY DETECTION	●	GNN	◎	LGNN [ICLR, 2019]
	◐	GNN	◎ ⛶	MRFasGCN [AAAI, 2019]
	○	GNN	◎ ⛶	NOCD [DLG, 2019]
	○	GNN	◎	AGC [IJCAI, 2019]
	○	GNN	◎	AGE [KDD, 2020]
	○	GNN, k-means	◎	CommDGI [CIKM, 2020]
	○	GNN	◎	DAEGC [IJCAI, 2019]
	○	GNN	◎	SDCN [WWW, 2020]
	○	GNN	◎	O2MAC [WWW, 2020]

● Supervised ◐ Semi-supervised ○ Unsupervised ⊡ Reinforcement ◎ Effectiveness ⏱ Efficiency ⛶ Scalability

Summary: ML for Subgraph

Subgraph Problem	Paradigm	Algorithm	Advance	Method
MAX COMMON SUBGRAPH	▣	Search + RL	⏱	McSplit+RL [AAAI, 2020]
	◑ ▣	GNN, Search + RL	🎯	GLSearch [ICML, 2020]
	●	GNN	🎯 ⏱	NeuralMCS [preprint, 2019]
SUBGRAPH ISOMORPHISM COUNTING	◑	GNN	⏱ ⤢	DIAMNet [SIGKDD, 2020]
	◑	GNN + Active Learning	⏱ ⤢	ALSS [SIGMOD, 2021]
	◑	GNN + Adversarial Learning	⏱ ⤢	NeurSC [SIGMOD, 2022]
	●	GNN	🎯	LRP [NIPS, 2020]
	●	GNN	🎯	RNP-GNN [arxiv, 2021]
	●	GNN	🎯	DMPNN [AAAI, 2022]
SUBGRAPH MATCHING	◑	Active Learning	⏱	ActiveMatch [ICBD, 2021]
	▣	GNN + RL	🎯 ⏱	RL-QVO [arxiv, 2022]
	●	GNN	🎯 ⏱	NeuroMatch [arxiv, 2020]
	●	GNN	🎯	DMPNN [AAAI, 2022]

● Supervised ◑ Semi-supervised ○ Unsupervised ▣ Reinforcement 🎯 Effectiveness ⏱ Efficiency ⤢ Scalability

Summary: Focus of Approaches

Subgraph Problem	Approach	Effectiveness	Efficiency	Scalability	Flexibility
Community Search	Non-ML	✓	✓	✓	
	ML	✓	✓	✓	✓
Community Detection	Non-ML	✓	✓	✓	
	ML	✓	✓	✓	✓
Maximum Common Subgraph	Non-ML	✓		✓	
	ML	✓	✓		✓
Subgraph Isomorphism Counting	Non-ML	✓	✓		
	ML	✓	✓	✓	✓

Summary: Pros and Cons of ML Approaches

Subgraph Problem	Flexibility	Efficiency	Training Data	Learning Cost
COMMUNITY SEARCH	🙂	🙂	😐	😢
COMMUNITY DETECTION	🙂	🙂	🙂	😢
MAXIMUM COMMON SUBGRAPH	🙂	🙂	😢	😢
SUBGRAPH ISOMORPHISM COUNTING	🙂	🙂	😐	😐

Future Directions

Explore More Models for Subgraph Problems

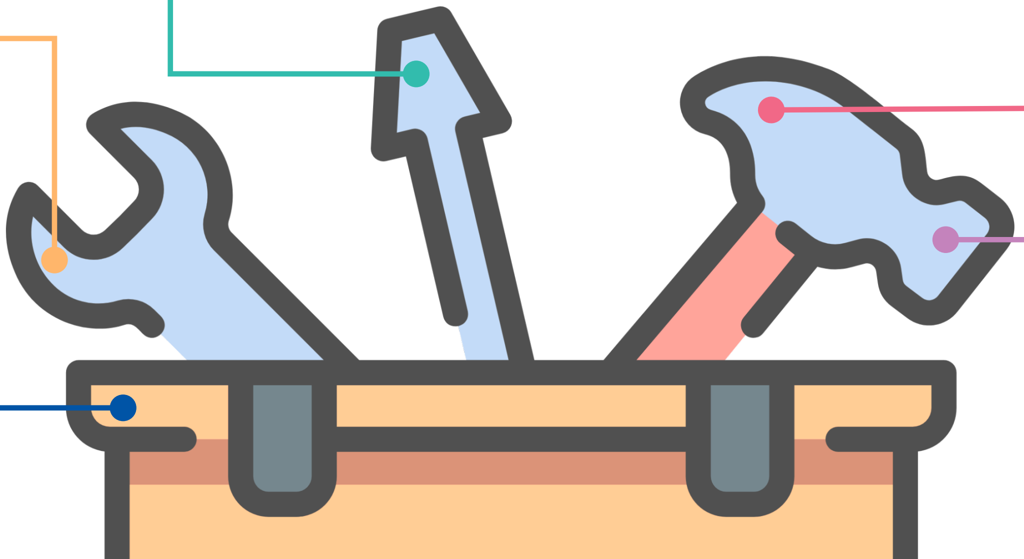
Self-Supervised
Generative, Active, Transfer

Reinforcement
Enumeration, Optimization

Semi-Supervised
Contrastive, Adversarial

Unsupervised
Embedding, Clustering

Supervised
Regression, Classification



Future Directions

Employ hybrid models of ML and non-ML approaches

Non-ML Approach

- Free-of-training
- Easy-explanation
- Mature strategy

ML Approach

- High Flexibility
- Better efficiency
- Model variety

Future Directions

Extend to other graph problems

DENSEST SUBGRAPH

Conventional: Network Flow

KClist++ [VLDB’20]: Sample & Search

B. Sun, et al. 2020. KClist++: a simple algorithm for finding k-clique densest subgraphs in large graphs. Proc. VLDB Endow. 13, 10, 1628–1640

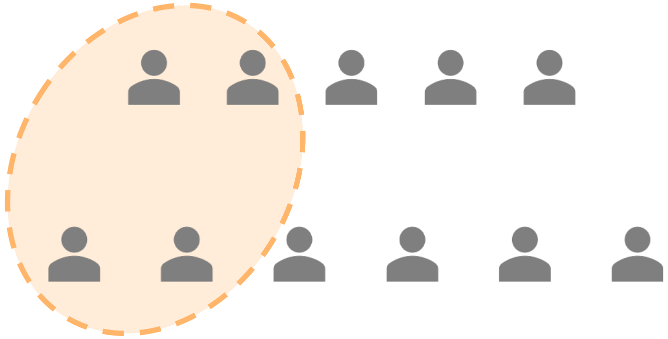
Ma et al [SIGMOD’22]: Convex Programming

C. Ma, et al. 2022. A Convex-Programming Approach for Efficient Directed Densest Subgraph Discovery. In SIGMOD'22, 845–859.

Future Directions

Extend to other graph problems

BIPARTITE SUBGRAPH



Conventional: BnB Search

BCList++ [VLDB’22]: Backtrack & Prune

J. Yang, Y. Peng, and W. Zhang. (p,q)-biclique Counting and Enumeration for Large Sparse Bipartite Graphs. PVLDB, 15(2): 141-153, 2022.

FastBB [SIGMOD’23]: Symmetric Branching

K. Yu and C. Long. 2023. Maximum k-Biplex Search on Bipartite Graphs: A Symmetric-BK Branching Approach. Proc. ACM Manag. Data.

Outline

THANK YOU

Slides available

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

香 港 大 學
THE UNIVERSITY OF HONG KONG